Int J Comput Vis (2016) 118:130–150
DOI 10.1007/s11263-015-0862-5

Exploiting Privileged Information from Web Data for Action and Event Recognition

Li Niu[1] · Wen Li[2] · Dong Xu[3]

Received: 15 October 2014 / Accepted: 23 September 2015 / Published online: 13 November 2015

Abstract In the conventional approaches for action and event recognition, sufficient labelled training videos are generally required to learn robust classifiers with good generalization capability on new testing videos. However, collecting labelled training videos is often time consuming and expensive. In this work, we propose new learning frameworks to train robust classifiers for action and event recognition by using freely available web videos as training data. We aim to address three challenging issues: (1) the training web videos are generally associated with rich textual descriptions, which are not available in test videos; (2) the labels of training web videos are noisy and may be inaccurate; (3) the data distributions between training and test videos are often considerably different. To address the first two issues, we propose a new framework called multi-instance learning with privileged information (MIL-PI) together with three new MIL methods, in which we not only take advantage of the additional textual descriptions of training web videos as privileged information, but also explicitly cope with noise in the loose labels of training web videos. When the training and test videos come from different data distributions, we further extend our MIL-PI as a new framework called domain adaptive MIL-PI. We also propose another three new domain adaptation methods, which can additionally reduce the data distribution mismatch between training and test videos. Comprehensive experiments for action and event recognition demonstrate the effectiveness of our proposed approaches.

Keywords Learning using privileged information · Multi-instance learning · Domain adaptation · Action recognition · Event recognition

1 Introduction

There is an increasing research interest in developing new action and event recognition technologies for a broad range of real-world applications including video search and retrieval, intelligent video surveillance and human computer interaction. While the two terms, actions and events, are often interchangeably used in several existing works (Aggarwal and Ryoo 2011; Bobick 1997), high-level events generally consist of a sequence of interactions or stand-alone actions (Jiang et al. 2013).

It is still a challenging computer vision task to recognize actions and events from videos due to considerable camera motion, cluttered backgrounds and large intra-class variations. Recently, a large number of approaches have been proposed for action recognition (Hu et al. 2009; Wang et al. 2011a; Yu et al. 2010; Zhu et al. 2009; Le et al. 2011; Lin et al. 2009; Shi et al. 2004; Zeng and Ji 2010; Wang et al. 2011b; Tran and Davis 2008; Morariu and Davis 2011) and event recognition (Chang et al. 2007; Xu and Chang 2008). Inter-

Communicated by M. Hebert.

✉ Wen Li
liwen@vision.ee.ethz.ch

Li Niu
lniu002@ntu.edu.sg

Dong Xu
dongxudongxu@gmail.com

[1] Interdisciplinary Graduate School, Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798, Singapore

[2] Computer Vision Laboratory, ETH Zurich, Sternwartstrasse 7, 8092 Zurich, Switzerland

[3] School of Electrical and Information Engineering, University of Sydney, Sydney, NSW 2006, Australia

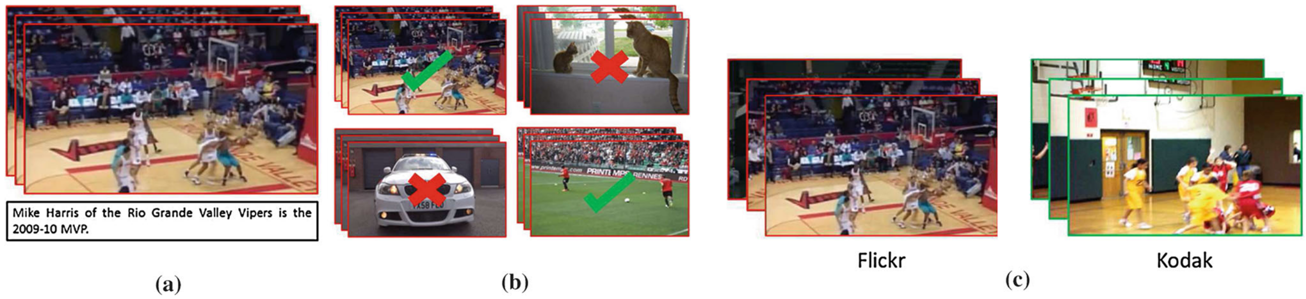
Mike Harris of the Rio Grande Valley Vipers is the 2009-10 MVP.
(a)
(b)
Flickr
(c)
Kodak

Fig. 1 Three challenging issues when learning from loosely labelled web videos: **a** the training web videos are additionally associated with rich textual descriptions, **b** the labels of relevant training web videos retrieved using the textual query "sports" are noisy, and **c** there is domain distribution mismatch between the training web videos and test videos

ested readers can refer to the recent surveys (Aggarwal and Ryoo 2011) and (Jiang et al. 2013) for more details. However, all the above methods follow the conventional approaches, in which a set of action/event lexicons are first defined and then a large corpus of training videos are collected with the action/event labels assigned by human annotators.

Collecting labelled training videos is often time-consuming and expensive. Meanwhile, rich and massive social media data are being posted to the video sharing websites like *Youtube* everyday, in which web videos are generally associated with valuable contextual information (e.g., tags, captions, and surrounding texts). Consequently, several recent works (Duan et al. 2012d; Chen et al. 2013a) were proposed to perform keywords (also called tags) based search to collect a set of relevant and irrelevant web videos, which are directly used as positive and negative training data for learning robust classifiers for action/event recognition. However, those works cannot effectively utilize the textual descriptions of training web videos because the test videos (e.g., the videos in the HMDB51 dataset) do not contain such textual descriptions.

In this work, we propose new learning frameworks for action and event recognition by using freely available web videos as training data. Specifically, as shown in Fig 1, we aim to address three challenging issues (1) the training web videos are usually accompanied with rich textual descriptions, while such textual descriptions are not available in the test videos; (2) the labels of training web videos are noisy (i.e., some labels are inaccurate); (3) the feature distributions of training and test videos may have very different statistical properties such as mean, intra-class variance and inter-class variance (Duan et al. 2012d, a).

To utilize the additional textual descriptions from the training web videos, we extract both visual features and textual features from the training videos. While we do not have textual features in the test videos, such textual features extracted from the training videos can still be used as privileged information, as shown in the recent work (Vapnik and Vashist 2009). Their work is motivated by human learning, where a teacher provides the students with hidden information through explanations, comments, comparisions, etc. (Vapnik and Vashist 2009). Similarly, we observe that the surrounding textual descriptions more or less describe the content of training data. So the textual features can additionally provide hidden information for learning robust classifiers by bridging the semantic gap between the low-level visual features and the high-level semantic concepts.

To cope with noisy labels of relevant training samples, we further employ the multi-instance learning (MIL) techniques because the MIL methods can still be used to learn classifiers even when the label of each training instance is unknown. Inspired by the recent works (Vijayanarasimhan and Grauman 2008; Li et al. 2011, 2012a), we first partition the training web videos into small subsets. By treating each subset as a "bag" and the videos in each bag as "instances", the MIL methods such as Sparse MIL (sMIL) (Bunescu and Mooney 2007), mi-SVM (Andrews et al. 2003) and MIL-CPB (Li et al. 2011) can be readily adopted to learn robust classifiers by using loosely labelled web videos as training data.

To address the first two challenging issues for action/event recognition, we propose our first framework called multi-instance learning with privileged information (MIL-PI). In this framework, we not only take advantage of the additional textual features from training web videos as privileged information, but also explicitly cope with noise in the loose labels of relevant training web videos. We also develop three new MIL approaches called sMIL-PI, mi-SVM-PI, and MIL-CPB-PI based on three existing MIL methods sMIL, mi-SVM and MIL-CPB, respectively. Moreover, we also observe that the action/event recognition performance could degrade when the training and test videos come from different data distributions, which is known as the *dataset bias* problem (Torralba and Efros 2011). To explicitly reduce the data distribution mismatch between the training and test videos, we further extend our MIL-PI framework by additionally introducing a Maximum Mean Discrepancy (MMD) based regularizer, which leads to our new MIL-PI-DA framework. We further extend sMIL-PI, mi-SVM-PI, and MIL-CPB-PI as sMIL-PI-DA, mi-SVM-PI-DA and MIL-CPB-PI-DA, respectively.

We conduct comprehensive experiments to evaluate our new approaches for action and event recognition. The results show that our newly proposed methods sMIL-PI, mi-SVM-PI and MIL-CPB-PI not only improve the existing MIL methods (i.e., sMIL, mi-SVM and MIL-CPB), but also outperform the learning methods using privileged information as well as other related baselines. Moreover, our newly proposed domain adaptation methods sMIL-PI-DA, mi-SVM-PI-DA and MIL-CPB-PI-DA are better than sMIL-PI, mi-SVM-PI and MIL-CPB-PI, respectively, and they also outperform the existing domain adaptation approaches.

In the preliminary conference version of this paper (Li et al. 2014b), we only discussed the bag-level MIL approaches sMIL-PI and sMIL-PI-DA for image categorization and image retrieval. In this work, we present a more general MIL-PI framework, which additionally incorporates the instance-level MIL methods, such as mi-SVM (Andrews et al. 2003) and MIL-CPB (Li et al. 2011). We also propose a new MIL-PI-DA framework and two more domain adaptation methods mi-SVM-PI-DA and MIL-CPB-PI-DA. Moreover, we additionally evaluate our newly proposed methods for action and event recognition on the benchmark datasets.

2 Related Work

2.1 Learning from Web Data

Researchers have proposed effective methods to employ massive web data for various computer vision applications (Schroff et al. 2011; Torralba et al. 2008; Fergus et al. 2005; Hwang and Grauman 2012). Torralba et al. (Torralba et al. 2008) used a nearest neighbor (NN) based approach for object and scene recognition by leveraging a large dataset with 80 million tiny images. Fergus et al. (2005) proposed a topic model based approach for object categorization by exploiting the images retrieved from Google image search, while Hwang and Grauman (2012) employed kernel canonical correlation analysis (KCCA) for image retrieval using different features. Recently, Chen et al. (2013b) proposed the NEIL system for automatically labeling instances and extracting the visual relationships.

Our work is more related to Vijayanarasimhan and Grauman (2008); Duan et al. (2011); Li et al. (2012a, b, 2011); Leung et al. (2011), which used multi-instance learning approaches to explicitly cope with noise in the loose labels of web images or web videos. In particular, those works first partitioned the training images into small subsets. By treating each subset as a “bag” and the images/videos in each bag as “instances”, they formulated this task as a multi-instance learning problem. The bag-based MIL method Sparse MIL as well as its variant were used in Vijayanarasimhan and Grauman (2008) for image categorization, while an instance-based approach called MIL-CPB was developed in Li et al. (2011) for image retrieval. Moreover, a weighted MIL-Boost approach was proposed in Leung et al. (2011) for video categorization. Besides the above multi-instance learning methods, some other approaches were also proposed to cope with label noise. For instance, Natarajan et al. (2013) proposed two approaches to modify the loss function for learning with noisy labels, in which the first approach uses the unbiased estimator of loss function and the second approach uses a weighted loss function. Bootkrajang and Kabán (2014) proposed a robust Multiple Kernel Logistic Regression algorithm (rMKLR), which incorporates the label flip probabilities in the loss function. However, the works in Vijayanarasimhan and Grauman (2008); Li et al. (2011); Leung et al. (2011); Natarajan et al. (2013); Bootkrajang and Kabán (2014) did not consider the additional features in training data, and thus they can only employ the visual features for learning MIL classifiers for action/event recognition.[1] In contrast, we propose a new action/event recognition framework MIL-PI by incorporating the additional textual features of training samples as privileged information.

2.2 Learning with Additional Information

Our approach is motivated by the work on learning using privileged information (LUPI) (Vapnik and Vashist 2009), in which training data contains additional features (i.e., privileged information) that are not available in the testing stage. Privileged information was also used for distance metric learning (Fouad et al. 2013), multiple task learning (Liang et al. 2009) and learning to rank (Sharmanska et al. 2013). However, all those works only considered the supervised learning scenario using training data with accurate supervision. In contrast, we formulate a new MIL-PI framework in order to cope with noise in the loose labels of relevant training web videos.

Our work is also related to attribute based approaches (Ferrari and Zisserman 2007; Farhadi et al. 2009), in which the attribute classifiers are learnt to extract the mid-level features. However, the mid-level features can be extracted from both training and testing images. Similarly, the classeme based approaches (Torresani et al. 2010; Li et al. 2013) were proposed to use the training images from additionally annotated

[1] The work in Li et al. (2011) used both visual and textual features in the training process. However, it also requires the textual features in the testing process.

concepts to obtain the mid-level features. Those methods can be readily applied to our application by using the mid-level features as the main features to replace our current visual features (i.e., the improved dense trajectory features (Wang and Schmid 2013) in our experiments). However, the additional textual features, which are not available in the testing samples, can still be used as privileged information in our MIL-PI framework. Moreover, those works did not explicitly reduce the data distribution mismatch between the training and testing samples as in our MIL-PI-DA framework.

2.3 Domain Adaptation

Our work is also related to the domain adaptation methods (Baktashmotlagh et al. 2013; Bergamo and Torresani 2010; Fernando et al. 2013; Huang et al. 2007; Gopalan et al. 2011; Gong et al. 2012; Kulis et al. 2011; Duan et al. 2012a; Bruzzone and Marconcini 2010; Duan et al. 2012d, c; Li et al. 2014a). Huang et al. (2007) proposed a two-step approach by re-weighting the source domain samples. For domain adaptation, Kulis et al. (2011) proposed a metric learning method by learning an asymmetric nonlinear transformation, while Gopalan et al. (2011) and Gong et al. (2012) interpolated intermediate domains. SVM based approaches (Duan et al. 2012a; Bruzzone and Marconcini 2010; Duan et al. 2012d, c) were also developed to reduce the data distribution mismatch. Some recent approaches aimed to learn a domain invariant subspace (Baktashmotlagh et al. 2013) or align two subspaces from both domains (Fernando et al. 2013). Bergamo and Torresani (2010) proposed a domain adaptation method which can cope with the loosely labelled training data. However, their method requires the labelled training samples from the target domain, which are not required in our domain adaptation framework MIL-PI-DA. Moreover, our MIL-PI-DA framework achieves the best results for action/event recognition when the training and testing samples are from different datasets.

3 Multi-instance Learning Using Privileged Information

For ease of presentation, in the remainder of this paper, we use a lowercase/uppercase letter in boldface to denote a vector/matrix (e.g., $\mathbf{a}$ denotes a vector and $\mathbf{A}$ denotes a matrix). The superscript $'$ denotes the transpose of a vector or a matrix. We denote $\mathbf{0}_n, \mathbf{1}_n \in \mathbb{R}^n$ as the n-dim column vectors of all zeros and all ones, respectively. For simplicity, we also use $\mathbf{0}$ and $\mathbf{1}$ instead of $\mathbf{0}_n$ and $\mathbf{1}_n$ when the dimension is obvious. Moreover, we use $\mathbf{A} \circ \mathbf{B}$ to denote the element-wise product between two matrices $\mathbf{A}$ and $\mathbf{B}$. The inequality $\mathbf{a} \leq \mathbf{b}$ means that $a_i \leq b_i$ for $i = 1, \ldots, n$.

3.1 Problem Statement

Our task is to learn robust classifiers for action/event recognition by using loosely labelled web videos. Given any action/event name, relevant and irrelevant web videos can be automatically collected as training data by using tag-based video retrieval. Those relevant (resp., irrelevant) videos can be used as positive (resp., negative) training samples for learning classifiers for action/event recognition. However, not all those relevant videos are semantically related to the action/event name, because the web videos are generally associated with noisy tags. Hence, we refer to those automatically collected web videos as loosely labelled web videos.

Moreover, although the test videos do not contain textual information, the additional textual features extracted from the training videos can still be used to improve the recognition performance. As shown in Vapnik and Vashist (2009), the additional features that are only available in training data can be utilized as privileged information to help learn more robust classifiers for the main features (i.e., the features that are available for both training and test data).

To this end, we propose a new learning framework called multi-instance learning using privileged information (MIL-PI) for action/event recognition, in which we not only take advantage of the additional textual descriptions (i.e., privileged information) in training data but also effectively cope with noise in the loose labels of relevant training videos.

In particular, to cope with label noise in training data, we partition the relevant and irrelevant web videos into bags as in the recent works (Vijayanarasimhan and Grauman 2008; Li et al. 2011; Leung et al. 2011). The training bags constructed from relevant samples are labelled as positive and those from irrelevant samples are labelled as negative. Let us represent the training data as $\{(\mathcal{B}_l, Y_l)|_{l=1}^{L}\}$, where $\mathcal{B}_l$ is a training bag, $Y_l \in \{+1, -1\}$ is the corresponding bag label, and L is the total number of training bags. Each training bag $\mathcal{B}_l$ consists of a number of training instances, i.e., $\mathcal{B}_l = \{(\mathbf{x}_i, \tilde{\mathbf{x}}_i, y_i)|_{i\in\mathcal{I}_l}\}$, where $\mathcal{I}_l$ is the set of indices for the instances inside $\mathcal{B}_l$, $\mathbf{x}_i$ is the visual feature vector extracted from the i-th web video, $\tilde{\mathbf{x}}_i$ is the corresponding textual feature extracted from its surrounding textual descriptions, $y_i \in \{+1, -1\}$ is the ground truth label that indicates whether the i-th video is semantically related to the action/event name. Note the ground truth label y_i is unknown. Without loss of generality, we assume the positive bags are the first L^+ training bags with a total number of n^+ training instances. The total number of training instances in all training bags is denoted as n .

In our framework, we use the generalized constraints for the MIL problem (Li et al. 2011). As shown in Li et al. (2011), the relevant samples usually contain a portion of positive samples, while it is more likely that the irrelevant samples are all negative samples. Namely, we have

$$\begin{cases} \sum_{i\in\mathcal{I}_l} \frac{y_i+1}{2} \geq \sigma|\mathcal{B}_l|, & \forall Y_l = 1, \\ y_i = -1, & \forall i \in \mathcal{I}_l \text{ and } Y_l = -1, \end{cases} \tag{1}$$

where $|\mathcal{B}_l|$ is the cardinality of the bag $\mathcal{B}_l$, and $\sigma > 0$ is a predefined ratio based on prior information. In other words, each positive bag is assumed to contain at least a portion of true positive instances, and all instances in a negative bag are assumed to be negative samples.

Recall the textual descriptions associated with the training videos are also noisy, so privileged information may not be always reliable as in Vapnik and Vashist (2009); Sharmanska et al. (2013). Considering the labels of instances in the negative bags are known to be negative (Vijayanarasimhan and Grauman 2008; Li et al. 2011), and the results after employing noisy privileged information for the instances in the negative bags are generally worse (see our experiments in Sect. 5.3), we only utilize privileged information for positive bags in our methods. However, it is worth mentioning that our method can be readily used to employ privileged information for the instances in all training bags.

In the following, we firstly introduce two LUPI approaches called SVM+ and partial SVM+ (pSVM+) that are related to this work. Then we propose a new bag-level MIL-PI method called sMIL-PI in Sect. 3.3 based on Sparse MIL (sMIL) (Bunescu and Mooney 2007), and also propose two instance-level MIL-PI methods called mi-SVM-PI and MIL-CPB-PI in Sect. 3.4 based on mi-SVM (Andrews et al. 2003) and MIL-CPB (Li et al. 2011), respectively.

3.2 Learning using Privileged Information

Let us denote the training data as $\{(\mathbf{x}_i, \tilde{\mathbf{x}}_i, y_i)|_{i=1}^n\}$, where $\mathbf{x}_i$ is main feature for the i-th training sample, $\tilde{\mathbf{x}}_i$ is the corresponding feature representation of privileged information which is not available for testing data, $y_i \in \{+1, -1\}$ is the class label, and n is the total number of training samples. Here the class label y_i of each training sample is assumed to be given. The goal of LUPI is to learn the classifier $f(\mathbf{x}) = \mathbf{w}'\phi(\mathbf{x}) + b$, where $\phi(\cdot)$ is a nonlinear feature mapping function. We also define another nonlinear feature mapping function $\tilde{\phi}(\cdot)$ for privileged information.

SVM+: SVM+ builds up the traditional SVM by further exploiting privileged information in training data. The objective of SVM+ is as follows,

$$\begin{aligned} \min_{\tilde{\mathbf{w}},\tilde{b},\mathbf{w},b} \quad & \frac{1}{2}\left(\|\mathbf{w}\|^2 + \gamma\|\tilde{\mathbf{w}}\|^2\right) + C\sum_{i=1}^{n} \xi(\tilde{\mathbf{x}}_i), \\ \text{s.t.} \quad & y_i(\mathbf{w}'\phi(\mathbf{x}_i) + b) \geq 1 - \xi(\tilde{\mathbf{x}}_i), \quad \xi(\tilde{\mathbf{x}}_i) \geq 0, \quad \forall i, \end{aligned} \tag{2}$$

where γ and C are the tradeoff parameters, $\xi(\tilde{\mathbf{x}}_i) = \tilde{\mathbf{w}}'\tilde{\phi}(\tilde{\mathbf{x}}_i) + \tilde{b}$ is the *slack function*, which replaces the slack variable $\xi_i \geq 0$ in the hinge loss in SVM. Such a slack function plays a role of teachers in the training process (Vapnik and Vashist 2009). Recall the slack variable ξ_i in SVM tells about how difficult to classify the training sample $\mathbf{x}_i$. The slack function $\xi(\tilde{\mathbf{x}}_i)$ is expected to model the optimal slack variable ξ_i by using privileged information, which is analogous to the comments and explanations from teachers in human learning (Vapnik and Vashist 2009). Similar to SVM, SVM+ can be solved in the dual form by optimizing a quadratic programming problem.

pSVM+: In some situations, privileged information may not be available for all the training samples. Particularly, when the training dataset contains l samples $\{(\mathbf{x}_i, \tilde{\mathbf{x}}_i, y_i)|_{i=1}^l\}$ with privileged information and $n-l$ samples $\{(\mathbf{x}_i, y_i)|_{i=l+1}^n\}$ without privileged information, the slack function can only be introduced for the l training samples with privileged information. We refer to this case of SVM+ as partial SVM+ or pSVM+ for short. According to Vapnik and Vashist (2009), we can formulate pSVM+ as follows:

$$\begin{aligned} \min_{\tilde{\mathbf{w}},\tilde{b},\mathbf{w},b,\boldsymbol{\eta}} \quad & \frac{1}{2}\left(\|\mathbf{w}\|^2 + \gamma\|\tilde{\mathbf{w}}\|^2\right) + C_1\sum_{i=1}^{l} \xi(\tilde{\mathbf{x}}_i) + \sum_{i=l+1}^{n} \eta_i, \\ \text{s.t.} \quad & y_i(\mathbf{w}'\phi(\mathbf{x}_i) + b) \geq 1 - \xi(\tilde{\mathbf{x}}_i), \quad \forall i = 1, \ldots, l, \\ & \xi(\tilde{\mathbf{x}}_i) \geq 0, \quad \forall i = 1, \ldots, l, \\ & y_i(\mathbf{w}'\phi(\mathbf{x}_i) + b) \geq 1 - \eta_i, \quad \forall i = l+1, \ldots, n, \\ & \eta_i \geq 0, \quad \forall i = l+1, \ldots, n, \end{aligned} \tag{3}$$

where γ and C_1 are the tradeoff parameters, $\xi(\tilde{\mathbf{x}}_i) = \tilde{\mathbf{w}}'\tilde{\phi}(\tilde{\mathbf{x}}_i) + \tilde{b}$ is the slack function, and $\boldsymbol{\eta} = [\eta_{l+1}, \ldots, \eta_n]'$ is the slack variable in the hinge loss. In fact, SVM+ can be treated as a special case of pSVM+ when $l = n$. Similar to SVM, pSVM+ can also be solved in the dual form by optimizing a quadratic programming problem.

3.3 Bag-level MIL using Privileged Information

The bag-level MIL methods (Chen et al. 2006; Bunescu and Mooney 2007) focus on the classification of bags. As the labels of training bags are known, by transforming each training bag to one training sample, the MIL problem becomes a supervised learning problem. Such a strategy can also be applied to our MIL-PI framework, and we refer to our new method as *sMIL-PI*.

3.3.1 sMIL-PI

Let us denote $\psi(\mathcal{B}_l)$ as the feature mapping function which converts a training bag into a single feature vector. The feature mapping function in sMIL is defined as the mean of instances inside the bag, i.e., $\psi(\mathcal{B}_l) = \frac{1}{|\mathcal{B}_l|}\sum_{i\in\mathcal{I}_l}\phi(\mathbf{x}_i)$, where $|\mathcal{B}_l|$ is the cardinality of the bag $\mathcal{B}_l$. Recall the labels for negative instances are assumed to be negative, so we only apply the feature mapping function on the positive training bags. For ease of presentation, we denote a set of virtual training samples $\{\mathbf{z}_j|_{j=1}^m\}$, in which $\mathbf{z}_1, \ldots, \mathbf{z}_{L^+}$ are the samples mapped from the positive bags $\{\psi(\mathcal{B}_j)|_{j=1}^{L^+}\}$, the remaining samples $\mathbf{z}_{L^++1}, \ldots, \mathbf{z}_m$ are the instances $\{\phi(\mathbf{x}_i)|i \in \mathcal{I}_l, Y_l = -1\}$ in the negative bags.

When there is additional privileged information for training data, we define another feature mapping function $\tilde{\psi}(\mathcal{B}_l)$ on each training bag as the mean of instances inside the bag by using privileged information, i.e., $\tilde{\mathbf{z}}_j = \tilde{\psi}(\mathcal{B}_j) = \frac{1}{|\mathcal{B}_j|}\sum_{i\in\mathcal{I}_j}\tilde{\phi}(\tilde{\mathbf{x}}_i)$ for $j = 1, \ldots, L^+$. Based on the SVM+ formulation, the objective of our sMIL-PI can be formulated as,

$$\min_{\mathbf{w},b,\tilde{\mathbf{w}},\tilde{b},\boldsymbol{\eta}} \frac{1}{2}\left(\|\mathbf{w}\|^2 + \gamma\|\tilde{\mathbf{w}}\|^2\right) + C_1\sum_{j=1}^{L^+}\xi(\tilde{\mathbf{z}}_j) + \sum_{j=L^++1}^{m}\eta_j,$$

$$\text{s.t.} \quad \mathbf{w}'\mathbf{z}_j + b \geq p_j - \xi(\tilde{\mathbf{z}}_j), \quad \forall j = 1, \ldots, L^+, \tag{4}$$

$$\mathbf{w}'\mathbf{z}_j + b \leq -1 + \eta_j, \quad \forall j = L^+ + 1, \ldots, m, \tag{5}$$

$$\xi(\tilde{\mathbf{z}}_j) \geq 0, \quad \forall j = 1, \ldots, L^+, \tag{6}$$

$$\eta_j \geq 0, \qquad \forall j = L^+ + 1, \ldots, m \tag{7}$$

where $\mathbf{w}$ and b are the variables of the classifier $f(\mathbf{z}) = \mathbf{w}'\mathbf{z} + b$, γ, C_1 are the tradeoff parameters, $\boldsymbol{\eta} = [\eta_{L^++1}, \ldots, \eta_m]'$, the slack function is defined as $\xi(\tilde{\mathbf{z}}_j) = \tilde{\mathbf{w}}'\tilde{\mathbf{z}}_j + \tilde{b}$, and p_j is the virtual label for the virtual sample $\mathbf{z}_j$. In sMIL (Bunescu and Mooney 2007), the virtual label is calculated by leveraging the instance labels of each positive bag. As sMIL assumes that there is at least one true positive sample in each positive bag, the virtual label of positive virtual sample $\mathbf{z}_j$ is $p_j = \frac{1-(|\mathcal{B}_j|-1)}{|\mathcal{B}_j|} = \frac{2-|\mathcal{B}_j|}{|\mathcal{B}_j|}$. Similarly, for our sMIL-PI using the generalized MIL constraints in (1), we can derive it as $p_j = \frac{\sigma|\mathcal{B}_j|-(1-\sigma)|\mathcal{B}_j|}{|\mathcal{B}_j|} = 2\sigma - 1$. Note the difference between (4) and pSVM+ is that we use the bag-level features instead of instance-level features and change the margin in the constraint from 1 to p_j.

By introducing dual variable $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_m]'$ for the constraints in (4) and (5), and also introducing dual variable $\boldsymbol{\beta} = [\beta_1, \ldots, \beta_{L^+}]'$ for the constraints in (6), respectively, we arrive at the dual from of (4) as follows,

$$\begin{aligned}\min_{\boldsymbol{\alpha},\boldsymbol{\beta}} \quad & -\mathbf{p}'\boldsymbol{\alpha} + \frac{1}{2}\boldsymbol{\alpha}'(\mathbf{K}\circ\mathbf{y}\mathbf{y}')\boldsymbol{\alpha} \\ & + \frac{1}{2\gamma}(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1})'\tilde{\mathbf{K}}(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1}), \\ \text{s.t.} \quad & \boldsymbol{\alpha}'\mathbf{y} = 0, \quad \mathbf{1}'(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1}) = 0, \\ & \bar{\boldsymbol{\alpha}} \leq \mathbf{1}, \quad \boldsymbol{\alpha} \geq \mathbf{0}, \quad \boldsymbol{\beta} \geq \mathbf{0},\end{aligned} \tag{8}$$

where $\hat{\boldsymbol{\alpha}} \in \mathbb{R}^{L^+}$ and $\bar{\boldsymbol{\alpha}} \in \mathbb{R}^{m-L^+}$ are from $\boldsymbol{\alpha} = [\hat{\boldsymbol{\alpha}}', \bar{\boldsymbol{\alpha}}']'$, $\mathbf{y} = [\mathbf{1}'_{L^+}, -\mathbf{1}'_{m-L^+}]'$ is the label vector, $\mathbf{p} = [p_1, \ldots, p_{L^+}, \mathbf{1}'_{m-L^+}]' \in \mathbb{R}^m$, $\mathbf{K} \in \mathbb{R}^{m\times m}$ is the kernel matrix constructed by using the visual features, $\tilde{\mathbf{K}} \in \mathbb{R}^{L^+\times L^+}$ is the kernel matrix constructed by using privileged information (i.e., the textual features). The above problem is jointly convex in $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, and can be solved by optimizing a quadratic programming problem.

3.4 Instance-level MIL using Privileged Information

Different from the bag-level MIL methods, the instance-level MIL methods (Andrews et al. 2003; Li et al. 2011) directly solve the classification problem for the instances. However, the labels of training instances are unknown, so one needs to infer the instance labels when learning the MIL classifier. Inspired by the works in Andrews et al. (2003); Li et al. (2011), we formulate the instance-level MIL-PI problem as follows,

$$\begin{aligned}\min_{\substack{\mathbf{y}\in\mathcal{Y}\\ \tilde{\mathbf{w}},\tilde{b},\mathbf{w},b,\boldsymbol{\eta}}} \quad & \frac{1}{2}\left(\|\mathbf{w}\|^2 + \gamma\|\tilde{\mathbf{w}}\|^2\right) \\ & + C_1\sum_{i=1}^{n^+}\xi(\tilde{\phi}(\tilde{\mathbf{x}}_i)) + \sum_{i=n^++1}^{n}\eta_i,\end{aligned} \tag{9}$$

$$\text{s.t.} \quad y_i(\mathbf{w}'\phi(\mathbf{x}_i) + b) \geq 1 - \xi(\tilde{\phi}(\tilde{\mathbf{x}}_i)), \tag{10}$$

$$\xi(\tilde{\phi}(\tilde{\mathbf{x}}_i)) \geq 0, \qquad i = 1, \ldots, n^+, \tag{11}$$

$$y_i(\mathbf{w}'\phi(\mathbf{x}_i) + b) \geq 1 - \eta_i, \tag{12}$$

$$\eta_i \geq 0, \qquad i = n^+ + 1, \ldots, n, \tag{13}$$

where $\mathcal{Y} = \{\mathbf{y}|\mathbf{y}$ satisfies the constraints in (1)$\}$ is the feasible set of labelings for training instances with $\mathbf{y} = [y_1, \ldots, y_n]'$ being a feasible label vector, $\boldsymbol{\eta} = [\eta_{n^++1}, \ldots, \eta_n]'$, γ and C_1 are the tradeoff parameters, and $\xi(\tilde{\phi}(\tilde{\mathbf{x}})) = \tilde{\mathbf{w}}'\tilde{\phi}(\tilde{\mathbf{x}}) + \tilde{b}$ is the slack function similarly as in sMIL-PI. The difference between (9) and pSVM+ is that the label vector $\mathbf{y}$ is also a variable which needs to be optimized in (9).

Note in this formulation, we need to infer the instance labels in the label vector $\mathbf{y}$, and simultaneously learn the classifier. It is a nontrivial mixed-integer programming problem, because the number of all possible labelings (i.e., $|\mathcal{Y}|$) increases exponentially w.r.t. the number of positive instances n^+. In mi-SVM (Andrews et al. 2003), an iterative approach is adopted to learn an SVM classifier and update the label vector $\mathbf{y}$ by using the prediction from the learnt classifier. In MIL-CPB (Li et al. 2011), a multiple kernel learning (MKL) based approach is proposed to learn an optimal kernel by optimizing the linear combination of the label kernels associated with all possible label vectors. We respectively apply those two strategies to our objective function

Algorithm 1 The optimization algorithm for solving the objective function of our mi-SVM-PI

```
Input: Training data $\{(\mathcal{B}_l, Y_l)|_{l=1}^{L}\}$ (see Sect. 3.1).
Initialize $\mathbf{y} = [\mathbf{1}'_{n^+}, -\mathbf{1}'_{n-n^+}]'$.
repeat
    Train $f(\mathbf{x}) = \mathbf{w}'\phi(\mathbf{x}) + b$ by solving a pSVM+ problem based on $\mathbf{y}$.
    Calculate the decision values of training instances by using the learnt $f(\mathbf{x})$.
    Based on the decision values, obtain $\mathbf{y}$ that satisfies the constraints in (1).
until The labeling vector $\mathbf{y}$ does not change.
Output: The learnt classifier $f(\mathbf{x}) = \mathbf{w}'\phi(\mathbf{x}) + b$.
```

in (9), and develop two instance-level MIL-PI approaches, mi-SVM-PI and MIL-CPB-PI.

3.4.1 mi-SVM-PI

In mi-SVM-PI, we adopt the strategy in mi-SVM (Andrews et al. 2003) and use the similar iterative updating approach to solve our instance based MIL-PI problem in (9). Specifically, as shown in Algorithm 1, we first initialize the label vector $\mathbf{y}$ by setting the labels of instances as their corresponding bag labels. Then we employ the alternating optimization method to iteratively solve a pSVM+ problem by using the current label vector $\mathbf{y}$, and infer $\mathbf{y}$ by using the learnt classifier $f(\mathbf{x}) = \mathbf{w}'\phi(\mathbf{x}) + b$ at the previous iteration. For any positive bag $\mathcal{B}_l$ where the constraint in (1) is not satisfied, we additionally set the labels of $\sigma|\mathcal{B}_l|$ instances with the largest decision values in this positive bag to be positive. The above process is repeated until $\mathbf{y}$ does not change.

3.4.2 MIL-CPB-PI

The instance-level MIL-PI formulation in (9) can also be solved by optimizing an MKL problem as in MIL-CPB, as discussed in Li et al. (2011). The main idea is to firstly relax the duality of (9) to its tight lower bound. Then we show that the relaxed problem shares a similar form with the MKL problem, and thus can be similarly optimized by solving a convex problem in the primal form.

To derive the solution of our MIL-CPB-PI method, we absorb the bias term b in (9) into $\mathbf{w}$ by augmenting the feature vector $\phi(\mathbf{x}_i)$ with an additional dimension with its value being 1 similarly as in Li et al. (2011). By respectively introducing the dual variables $\hat{\boldsymbol{\alpha}} \in \mathbb{R}^{n^+}$, $\bar{\boldsymbol{\alpha}} \in \mathbb{R}^{n-n^+}$ and $\boldsymbol{\beta} \in \mathbb{R}^{n^+}$ for the constraints in (10), (12), and (11), and defining $\boldsymbol{\alpha} = [\hat{\boldsymbol{\alpha}}', \bar{\boldsymbol{\alpha}}']' \in \mathbb{R}^n$, we arrive at the dual problem of (9) as follows,

$$\min_{\mathbf{y}\in\mathcal{Y}} \max_{(\boldsymbol{\alpha},\boldsymbol{\beta})\in\mathcal{S}} \quad \mathbf{1}'\boldsymbol{\alpha} - \frac{1}{2}\boldsymbol{\alpha}'(\mathbf{Q}\circ\mathbf{y}\mathbf{y}')\boldsymbol{\alpha} \\ -\frac{1}{2\gamma}(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1})'\tilde{\mathbf{K}}(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1}), \tag{14}$$

where $\mathbf{Q} = \mathbf{K} + \mathbf{1}\mathbf{1}'$ with $\mathbf{K} \in \mathbb{R}^{n\times n}$ being the kernel matrix constructed by using the visual features, $\tilde{\mathbf{K}} \in \mathbb{R}^{n^+\times n^+}$ is the kernel matrix constructed by using the textual features, $\mathcal{S} = \{(\boldsymbol{\alpha}, \boldsymbol{\beta})|\mathbf{1}'(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1}) = 0, \bar{\boldsymbol{\alpha}} \le \mathbf{1}, \boldsymbol{\alpha} \ge \mathbf{0}, \boldsymbol{\beta} \ge \mathbf{0}\}$ is the feasible set.

Note that each label vector $\mathbf{y}$ forms a label kernel $\mathbf{Q}\circ\mathbf{y}\mathbf{y}'$ in the duality in (14). Inspired by Li et al. (2011), instead of directly optimizing an optimal label kernel $\mathbf{Q}\circ\mathbf{y}\mathbf{y}'$, we seek for an optimal linear combination of all possible label kernels. We write the relaxed problem as follows,

$$\min_{\mathbf{d}\in\mathcal{D}} \max_{(\boldsymbol{\alpha},\boldsymbol{\beta})\in\mathcal{S}} \quad \mathbf{1}'\boldsymbol{\alpha} - \frac{1}{2}\boldsymbol{\alpha}'\left(\sum_{t=1}^{T} d_t\mathbf{Q}\circ\mathbf{y}_t\mathbf{y}_t'\right)\boldsymbol{\alpha} \\ -\frac{1}{2\gamma}(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1})'\tilde{\mathbf{K}}(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1}), \tag{15}$$

where $\mathbf{y}_t \in \mathcal{Y}$ is the t-th label vector in the feasible set $\mathcal{Y}$, $T = |\mathcal{Y}|$ is the total number of label vectors in $\mathcal{Y}$, d_t is the combination coefficient of the label kernel $\mathbf{Q}\circ\mathbf{y}_t\mathbf{y}_t'$, $\mathbf{d} = [d_1, \ldots, d_T]'$ is the vector which contains all the combination coefficients, and $\mathcal{D} = \{\mathbf{d}|\mathbf{d}'\mathbf{1} = 1, \mathbf{d} \ge \mathbf{0}\}$ is the feasible set of $\mathbf{d}$.

Intuitively, for the optimization problem in (14), we search for an optimal $\mathbf{y}\mathbf{y}'$ in $\mathcal{Y}$, which is a set of discrete points in the space $\mathbb{R}^{n\times n}$. The optimization problem in (14) is a Mixed Integer Programming (MIP) problem and is NP-hard. In contrast, the optimization problem in (15) is in the convex hull of all possible $\mathbf{y}_t\mathbf{y}_t'$'s in $\mathbb{R}^{n\times n}$ (Li et al. 2009), which is a continuous region and makes the problem easier to be solved. Actually, by considering each $(\mathbf{Q}\circ\mathbf{y}_t\mathbf{y}_t')$ as a base kernel, the optimization problem in (15) shares a similar form with the MKL problem, which can be solved by optimizing a convex optimization problem in its primal problem (Kloft et al. 2011).

The main challenge for applying the existing MKL techniques to solve (15) is that we have too many base kernels, i.e., $T = |\mathcal{Y}|$ is possibly exponential to the number of positive instances n^+. Inspired by Infinite Kernel Learning (IKL) (Gehler and Nowozin 2008), we employ the cutting-plane algorithm to solve it. Specifically, by introducing a dual variable τ for the constraint $\mathbf{d}'\mathbf{1} = 1$ in $\mathcal{D}$, we arrive at the duality of (15) as follows (see the detailed derivations in Appendix

Algorithm 2 Approximately find the most violated y_t

```
Initialize $y_i = 1$ for all instances in positive bags $\{(\mathcal{B}_l, Y_l)|_{l=1}^{L^+}\}$.
repeat
    for each positive bag $\mathcal{B}_l$ do
        Fix the labeling of all the other positive bags, find the optimal instance labels for $\mathcal{B}_l$ that maximizes (17) by enumerating all the feasible instance labels for $\mathcal{B}_l$.
    end for
until no labels are changed.
```

Algorithm 3 The optimization algorithm for solving the objective function of our MIL-CPB-PI

```
Input: Training data $\{(\mathcal{B}_l, Y_l)|_{l=1}^{L}\}$ (see Sect. 3.1).
Initialize $\mathcal{C} = \{\mathbf{y}_0\}$ with $\mathbf{y}_0 = [\mathbf{1}'_{n^+}, -\mathbf{1}'_{n-n^+}]'$, and set $r = 0$.
repeat
    Set $r \leftarrow r + 1$.
    Based on $\mathcal{Y} = \mathcal{C}$, solve for $(\mathbf{d}, \boldsymbol{\alpha}, \boldsymbol{\beta})$ by optimizing the MKL problem in (15) (See Appendix 2 for the detailed solution).
    Set $\mathcal{C} \leftarrow \mathcal{C} \bigcup \mathbf{y}_r$ where $\mathbf{y}_r$ is obtained by solving (17).
until The objective of (15) converges.
Output: The learnt classifier $f(\mathbf{x})$.
```

1),

$$\max_{\tau, (\boldsymbol{\alpha}, \boldsymbol{\beta}) \in \mathcal{S}} \mathbf{1}'\boldsymbol{\alpha} - \frac{1}{2\gamma}(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1})'\tilde{\mathbf{K}}(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1}) - \tau,$$
$$\text{s.t.} \quad \frac{1}{2}\boldsymbol{\alpha}'\left(\mathbf{Q} \circ \mathbf{y}_t\mathbf{y}_t'\right)\boldsymbol{\alpha} \leq \tau, \quad \forall t = 1, \ldots, T. \tag{16}$$

As each of the constraints in (16) corresponds to a base kernel $\mathbf{Q} \circ \mathbf{y}_t\mathbf{y}_t'$, there are many constraints (i.e., $T = |\mathcal{Y}|$) in the above problem. The main idea of the cutting-plane algorithm is to approximate (16) by using only a few constraints. Specifically, we start from one constraint, and solve for $(\boldsymbol{\alpha}, \boldsymbol{\beta})$ and τ. If there is any constraint that cannot be satisfied, we add this constraint into the current optimization problem, and resolve for $(\boldsymbol{\alpha}, \boldsymbol{\beta})$ and τ again. The above process is repeated until all constraints are satisfied.

To find the violated constraint, we maximize the left-hand side of the constraint in (16), which can be written as follows,

$$\max_{\mathbf{y} \in \mathcal{Y}} \mathbf{y}'(\mathbf{Q} \circ \boldsymbol{\alpha}\boldsymbol{\alpha}')\mathbf{y}, \tag{17}$$

The above optimization problem approximately by enumerating the instance labels in a bag-by-bag fashion when the size of each bag is not too large, as discussed in Algorithm 2.

The algorithm of our MIL-CPB-PI is listed in Algorithm 3. We first initialize the labeling set as $\mathcal{C} = \{\mathbf{y}_0\}$. Then we iteratively train an MKL classifier by solving (15) based on $\mathcal{Y} = \mathcal{C}$ and update the labeling set $\mathcal{C}$ by adding the violated $\mathbf{y}_r$, which is obtained by solving (17) based on the current $\boldsymbol{\alpha}$. This process is repeated until the objective of (15) converges.

As we only need to solve an MKL problem based on a small set of base kernels at each iteration, the optimization procedure is much more efficient. It can be solved similarly as in the existing MKL solver in Kloft et al. (2011). We also give the detailed optimization procedure in Appendix 2.

Moreover, the objective of (15) decreases monotonously as r increases, because the labeling set is enlarged at each iteration. The final classifier can be presented as $f(\mathbf{x}) = \mathbf{w}'\phi(\mathbf{x}) + b$ with $\mathbf{w} = \sum_{i=1}^{n} \alpha_i \tilde{y}_i \phi(\mathbf{x}_i)$, where α_i is the i-th entry in the final dual variable $\boldsymbol{\alpha}$, and $\tilde{y}_i = \sum_{t=1}^{r} d_t y_{t,i}$ with $y_{t,i}$ being the i-th entry of $\mathbf{y}_t$.

4 Domain Adaptive MIL-PI

The training web videos often have very different statistical properties from the test videos, which is also known as the dataset bias problem (Torralba and Efros 2011). To reduce the domain distribution mismatch, we proposed a new domain adaptation framework by re-weighting the source domain samples when learning the classifiers. In the following, we develop our domain adaptation framework, which is referred to as MIL-PI-DA. Moreover, we also extend sMIL-PI (resp., mi-SVM-PI, MIL-CPB-PI) to sMIL-PI-DA (resp., mi-SVM-PI-DA, MIL-CPB-PI-DA).

Our work is inspired by the Kernel Mean Matching (KMM) method (Huang et al. 2007), in which the source domain samples are reweighted by minimizing the Maximum Mean Discrepancy (MMD) between two domains. However, KMM is a two-stage method, in which they first learn the weights for the source domain samples and then utilize the learnt weights to train a weighted SVM. Though the recent work (Chu et al. 2013) proposed to combine the primal formulation of weighted-SVM and a regularizer based on the MMD criterion, their objective function is non-convex, and thus the global optimal solution cannot be guaranteed. To this end, we propose a convex formulation by adding the regularizer based on the MMD criterion to the dual formulation of our MIL-PI framework, which leads to a convex objective function as discussed in Sect. 4.1. Formally, let us denote the target domain samples as $\{\mathbf{x}_i^t|_{i=1}^{n_t}\}$, and denote $\phi(\mathbf{x}_i^t)$ as the corresponding nonlinear feature. To distinguish the two domains, we append a superscript s to the source domain samples, i.e., $\{\mathbf{x}_i^s|_{i=1}^{n_s}\}$ and denote $\phi(\mathbf{x}_i^s)$ as the corresponding nonlinear feature.

4.1 Bag-Level Domain Adaptive MIL-PI

We propose a bag-level domain adaptive MIL-PI method sMIL-PI-DA, which is extended from sMIL-PI. We denote the objective in (8) as $H(\boldsymbol{\alpha}, \boldsymbol{\beta}) = -\mathbf{p}'\boldsymbol{\alpha} + \frac{1}{2}\boldsymbol{\alpha}'(\mathbf{K} \circ \mathbf{y}\mathbf{y}')\boldsymbol{\alpha} + \frac{1}{2\gamma}(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1})'\tilde{\mathbf{K}}(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1})$, and also denote the weights for source domain samples as $\boldsymbol{\theta} = [\theta_1, \ldots, \theta_m]'$ with each θ_i being the weight for the i-th source domain sample. We also denote $\{\mathbf{z}_i^s|_{i=1}^{m}\}$ (resp., $\{\mathbf{z}_i^t|_{i=1}^{n_t}\}$) as the set of virtual samples in the source (resp., target) domain, which are used in our sMIL-PI-DA. Note that $\mathbf{z}_i^s$'s and $\mathbf{z}_i^t$'s denote the visual features. Then, we formulate our sMIL-PI-DA method as follows,

$$\min_{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\theta}} \quad H(\boldsymbol{\alpha},\boldsymbol{\beta}) + \frac{\mu}{2}\|\frac{1}{m}\sum_{i=1}^{m}\theta_i\mathbf{z}_i^s - \frac{1}{n_t}\sum_{i=1}^{n_t}\mathbf{z}_i^t\|^2 \tag{18}$$

$$\text{s.t.} \quad \boldsymbol{\alpha}'\mathbf{y} = 0, \quad \mathbf{1}'(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1}) = 0, \tag{19}$$

$$\bar{\boldsymbol{\alpha}} \leq \mathbf{1}, \quad \boldsymbol{\beta} \geq \mathbf{0} \tag{20}$$

$$\mathbf{0} \leq \boldsymbol{\alpha} \leq C_2\boldsymbol{\theta}, \quad \mathbf{1}'\boldsymbol{\theta} = m, \tag{21}$$

where C_2 is a parameter and θ_i is the weight for $\mathbf{z}_i^s$. The last term in (18) is a regularizer based on the MMD criterion, which aims to reduce the domain distribution mismatch between two domains by reweighting the source domain samples as in KMM, and the constraints in (19) and (20) are from sMIL-PI. Note in (21), we use the box constraint $\mathbf{0} \leq \boldsymbol{\alpha} \leq C_2\boldsymbol{\theta}$ to regularize the dual variable $\boldsymbol{\alpha}$, which is similarly used in weighted SVM (Huang et al. 2007). In (21), the second constraint $\mathbf{1}'\boldsymbol{\theta} = m$ is used to enforce the expectation of sample weights to be 1. The problem in (18) is jointly convex with respect to $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$, and thus we can obtain the global optimum by optimizing a quadratic programming problem.

Interestingly, the primal form of (18) is closely related to the formulation of SVM+, as described below,

Proposition 1 *The primal form of (18) is equivalent to the following problem,*

$$\min_{\mathbf{w},b,\tilde{\mathbf{w}},\tilde{b},\hat{\mathbf{w}},\hat{b},\boldsymbol{\eta}} \quad J(\mathbf{w},b,\tilde{\mathbf{w}},\tilde{b},\boldsymbol{\eta}) + \frac{\lambda}{2}\|\hat{\mathbf{w}} - \rho\mathbf{v}\|^2 + C_2\sum_{i=1}^{m}\zeta(\mathbf{z}_i^s), \tag{22}$$

$$\text{s.t.} \quad \mathbf{w}'\mathbf{z}_i^s + b \geq p_i - \xi(\tilde{\mathbf{z}}_i^s) - \zeta(\mathbf{z}_i^s), \quad \forall i = 1,\ldots,L^+, \tag{23}$$

$$\mathbf{w}'\mathbf{z}_i^s + b \leq -1 + \eta_i + \zeta(\mathbf{z}_i^s), \quad \forall i = L^+ + 1,\ldots,m, \tag{24}$$

$$\xi(\tilde{\mathbf{z}}_i^s) \geq 0, \quad \forall i = 1,\ldots,L^+, \tag{25}$$

$$\eta_i \geq 0, \quad \forall i = L^+ + 1,\ldots,m, \tag{26}$$

$$\zeta(\mathbf{z}_i^s) \geq 0, \quad \forall i = 1,\ldots,m, \tag{27}$$

where $J(\mathbf{w},b,\tilde{\mathbf{w}},\tilde{b},\boldsymbol{\eta}) = \frac{1}{2}\left(\|\mathbf{w}\|^2 + \gamma\|\tilde{\mathbf{w}}\|^2\right) + C_1\sum_{j=1}^{L^+}\xi(\tilde{\mathbf{z}}_j^s) + \sum_{j=L^++1}^{m}\eta_j$ *is the objective function in (4),* $\zeta(\mathbf{z}_i^s) = \hat{\mathbf{w}}'\mathbf{z}_i^s + \hat{b}$, $\mathbf{v} = \frac{1}{m}\sum_{i=1}^{m}\mathbf{z}_i^s - \frac{1}{n_t}\sum_{i=1}^{n_t}\mathbf{z}_i^t$, $\lambda = \frac{(mC_2)^2}{\mu}$ *and* $\rho = \frac{mC_2}{\lambda}$.

The detailed proof is provided in Appendix 3.

Compared with the objective function in (4), we introduce one more slack function $\zeta(\mathbf{z}_i^s) = \hat{\mathbf{w}}'\mathbf{z}_i^s + \hat{b}$, and also regularize the weight vector of this slack function by using the regularizer $\|\hat{\mathbf{w}} - \rho\mathbf{v}\|^2$. Recall that the witness function in MMD is defined as $g(\mathbf{z}) = \frac{1}{\|\mathbf{v}\|}\mathbf{v}'\mathbf{z}$ (Gretton et al. 2012), which can be deemed as the mean similarity between $\mathbf{z}$ and the source domain samples (i.e., $\frac{1}{m}\sum_{i=1}^{m}\mathbf{z}_i^{s\prime}\mathbf{z}$) minus the mean similarity between $\mathbf{z}$ and the target domain samples (i.e., $\frac{1}{n_t}\sum_{i=1}^{n_t}\mathbf{z}_i^{t\prime}\mathbf{z}$). In other words, we conjecture that the witness function outputs a lower value when the sample $\mathbf{z}$ is closer to the target domain samples and vice versa. By using the regularizer $\|\hat{\mathbf{w}} - \rho\mathbf{v}\|^2$, we expect the new slack function $\zeta(\mathbf{z}_i^s) = \hat{\mathbf{w}}'\mathbf{z}_i^s + \hat{b}$ shares the similar trend[2] with the witness function $g(\mathbf{z}_i^s) = \frac{1}{\|\mathbf{v}\|}\mathbf{v}'\mathbf{z}_i^s$. As a result, the training error of the training sample $\mathbf{z}_i^s$ (i.e., $\xi(\tilde{\mathbf{z}}_i^s) + \zeta(\mathbf{z}_i^s)$ for the samples in the positive bags or $\eta_i + \zeta(\mathbf{z}_i^s)$ for the negative samples) will tend to be lower if it is closer to the target domain, which is helpful for learning a more robust classifier to better predict the target domain samples.

4.2 Instance-Level Domain Adaptive MIL-PI

Besides the bag-level MIL method sMIL, we can also incorporate the instance-level MIL methods, mi-SVM and MIL-CPB, into our MIL-PI-DA framework. We refer to our new approaches as mi-SVM-PI-DA and MIL-CPB-PI-DA, respectively.

4.2.1 mi-SVM-PI-DA

To derive the formulation of mi-SVM-PI-DA, we firstly write the duality of the mi-SVM-PI problem in (9) as follows,

$$\begin{aligned}\min_{\mathbf{y}\in\mathcal{Y}}\max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \quad & J(\boldsymbol{\alpha},\boldsymbol{\beta},\mathbf{y}) \doteq \mathbf{1}'\boldsymbol{\alpha} - \frac{1}{2}\boldsymbol{\alpha}'(\mathbf{K}\circ\mathbf{y}\mathbf{y}')\boldsymbol{\alpha} \\ & -\frac{1}{2\gamma}(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1})'\tilde{\mathbf{K}}(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1}), \\ \text{s.t.} \quad & \boldsymbol{\alpha}'\mathbf{y} = 0, \quad \mathbf{1}'(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1}) = 0, \\ & \bar{\boldsymbol{\alpha}} \leq \mathbf{1}, \quad \boldsymbol{\alpha} \geq \mathbf{0}, \quad \boldsymbol{\beta} \geq \mathbf{0}.\end{aligned} \tag{28}$$

where $\boldsymbol{\alpha} = [\hat{\boldsymbol{\alpha}}', \bar{\boldsymbol{\alpha}}']'$ and $\boldsymbol{\beta}$ are the dual variables defined similarly as in the duality of MIL-CPB-PI in (14).

Similarly as in sMIL-PI-DA, we also introduce the MMD based regularizer to (28) in order to reduce the domain distribution mismatch, which leads to our mi-SVM-PI-DA problem as follows,

[2] The bias term $\hat{b}$ and the scalar terms ρ and $\frac{1}{\|\mathbf{v}\|}$ will not change the trend of functions.

Algorithm 4 The optimization algorithm for solving the objective function of our mi-SVM-PI-DA

```
Input: Training data $\{(\mathcal{B}_l, Y_l)|_{l=1}^{L}\}$ (see Sect. 3.1).
Initialize $\mathbf{y} = [\mathbf{1}'_{n^+}, -\mathbf{1}'_{n_s-n^+}]'$.
repeat
    Train $f(\mathbf{x}) = \mathbf{w}'\phi(\mathbf{x}) + b$ by solving the subproblem in (30) based on $\mathbf{y}$.
    Calculate the decision values of training instances by using the learnt $f(\mathbf{x})$.
    Based on the decision values, obtain $\mathbf{y}$ that satisfies the constraints in (1).
until The labeling vector $\mathbf{y}$ does not change.
Output: The learnt classifier $f(\mathbf{x}) = \mathbf{w}'\phi(\mathbf{x}) + b$.
```

$$\begin{aligned}\min_{\mathbf{y}\in\mathcal{Y}}\max_{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\theta}} \quad & J(\boldsymbol{\alpha},\boldsymbol{\beta},\mathbf{y}) - \frac{\mu}{2}\|\frac{1}{n_s}\sum_{i=1}^{n_s}\theta_i\phi(\mathbf{x}_i^s) - \frac{1}{n_t}\sum_{i=1}^{n_t}\phi(\mathbf{x}_i^t)\|^2 \\ \text{s.t.} \quad & \boldsymbol{\alpha}'\mathbf{y} = 0, \quad \mathbf{1}'(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1}) = 0, \\ & \bar{\boldsymbol{\alpha}} \le \mathbf{1}, \quad \boldsymbol{\beta} \ge \mathbf{0}, \\ & \mathbf{0} \le \boldsymbol{\alpha} \le C_2\boldsymbol{\theta}, \quad 1'\boldsymbol{\theta} = n_s, \end{aligned} \tag{29}$$

where n_s is the number of source domain samples, n_t is the number of target domain samples. Similarly as in weighted SVM, the box constraint $\mathbf{0} \le \boldsymbol{\alpha} \le C_2\boldsymbol{\theta}$ is used, in which each θ_i is the weight for the i-th source domain sample. Note we minus the MMD based regularizer in (29), as the inner optimization problem is a maximization problem.

Similarly as in mi-SVM-PI, we solve the optimization problem in (29) in an iterative approach. When the label vector $\mathbf{y}$ is fixed, the subproblem can be written as,

$$\begin{aligned}\min_{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\theta}} \quad & -J(\boldsymbol{\alpha},\boldsymbol{\beta},\mathbf{y}) + \frac{\mu}{2}\left(\frac{1}{n_s^2}\boldsymbol{\theta}'\mathbf{K}\boldsymbol{\theta} - \frac{2}{n_s n_t}\boldsymbol{\theta}'\mathbf{K}_{st}\mathbf{1}\right) \\ \text{s.t.} \quad & \boldsymbol{\alpha}'\mathbf{y} = 0, \quad \mathbf{1}'(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1}) = 0, \\ & \bar{\boldsymbol{\alpha}} \le \mathbf{1}, \quad \boldsymbol{\beta} \ge \mathbf{0}, \\ & \mathbf{0} \le \boldsymbol{\alpha} \le C_2\boldsymbol{\theta}, \quad 1'\boldsymbol{\theta} = n_s, \end{aligned} \tag{30}$$

where $\mathbf{K}_{st} \in \mathbb{R}^{n_s \times n_t}$ is the kernel matrix measuring the similarity between the training samples and test samples by using visual features.

We describe the algorithm for solving mi-SVM-PI-DA in Algorithm 4. We first initialize the label vector $\mathbf{y}$ by setting the labels of instances as their corresponding bag labels. Then we iteratively solve the inner optimization problem based on the current $\mathbf{y}$, and infer $\mathbf{y}$ by using the learnt classifier $f(\mathbf{x}) = \mathbf{w}'\phi(\mathbf{x}) + b$ from the previous iteration. The inner optimization problem can be solved by optimizing a convex quadratic programming problem as in (30). For any positive bag $\mathcal{B}_l$ where the constraint in (1) is not satisfied, we additionally set the labels of $\sigma|\mathcal{B}_l|$ instances with the largest decision values in this positive bag to be positive. The above process is repeated until $\mathbf{y}$ does not change.

Similarly as sMIL-PI-DA, the primal form of (29) is also related to the formulation of SVM+, as described in Proposition 2 below,

Proposition 2 *The primal form of (29) is equivalent to the following problem,*

$$\begin{aligned}\min_{\substack{\mathbf{y},\mathbf{w},b,\tilde{\mathbf{w}},\\ \tilde{b},\hat{\mathbf{w}},\hat{b},\boldsymbol{\eta}}} \quad & J(\mathbf{w}, b, \tilde{\mathbf{w}}, \tilde{b}, \boldsymbol{\eta}) + \frac{\lambda}{2}\|\hat{\mathbf{w}} - \rho\mathbf{v}\|^2 \\ & +C_2\sum_{i=1}^{n_s}\zeta(\phi(\mathbf{x}_i^s)), && (31) \\ s.t. \quad & y_i(\mathbf{w}'\phi(\mathbf{x}_i^s) + b) \ge 1 - \xi(\tilde{\phi}(\tilde{\mathbf{x}}_i^s)) - \zeta(\phi(\mathbf{x}_i^s)), \\ & \forall i = 1, \ldots, n^+, && (32) \\ & y_i(\mathbf{w}'\phi(\mathbf{x}_i^s) + b) \ge 1 - \eta_i - \zeta(\phi(\mathbf{x}_i^s)), \\ & \forall i = n^+ + 1, \ldots, n_s, && (33) \\ & \xi(\tilde{\phi}(\tilde{\mathbf{x}}_i^s)) \ge 0, \quad \forall i = 1, \ldots, n^+, && (34) \\ & \eta_i \ge 0, \qquad \forall i = n^+ + 1, \ldots, n_s, && (35) \\ & \zeta(\phi(\mathbf{x}_i^s)) \ge 0, \quad \forall i = 1, \ldots, n_s, && (36) \end{aligned}$$

where $J(\mathbf{w}, b, \tilde{\mathbf{w}}, \tilde{b}, \boldsymbol{\eta}) = \frac{1}{2}\left(\|\mathbf{w}\|^2 + \gamma\|\tilde{\mathbf{w}}\|^2\right) + C_1\sum_{j=1}^{n^+}\xi(\tilde{\phi}(\tilde{\mathbf{x}}_j^s)) + \sum_{j=n^++1}^{n}\eta_j$ *is the objective function in (9),* $\zeta(\phi(\mathbf{x}_i^s)) = \hat{\mathbf{w}}'\phi(\mathbf{x}_i^s) + \hat{b}$, $\mathbf{v} = \frac{1}{n_s}\sum_{i=1}^{n_s}\phi(\mathbf{x}_i^s) - \frac{1}{n_t}\sum_{i=1}^{n_t}\phi(\mathbf{x}_i^t)$, $\lambda = \frac{(n_s C_2)^2}{\mu}$ *and* $\rho = \frac{n_s C_2}{\lambda}$.

We can similarly explain the regularizer $\frac{\lambda}{2}\|\hat{\mathbf{w}} - \rho\mathbf{v}\|^2$ and $\zeta(\phi(\mathbf{x}_i^s))$ as those for Proposition 1. It can also be similarly proved and the details are ignored here.

4.2.2 MIL-CPB-PI-DA

Let us denote the objective of the duality of MIL-CPB-PI in (15) as $J(\boldsymbol{\alpha},\boldsymbol{\beta},\mathbf{d}) = \mathbf{1}'\boldsymbol{\alpha} - \frac{1}{2}\boldsymbol{\alpha}'\left(\sum_{t=1}^{T} d_t\mathbf{Q} \circ \mathbf{y}_t\mathbf{y}_t'\right)\boldsymbol{\alpha} - \frac{1}{2\gamma}(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1})'\tilde{\mathbf{K}}(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1})$. Similarly, we can reduce the domain distribution mismatch by using a MMD based regularizer. We arrive at the objective function of our MIL-CPB-PI-DA as follows,

$$\begin{aligned}\min_{\mathbf{d}\in\mathcal{D}}\max_{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\theta}} \quad & J(\boldsymbol{\alpha},\boldsymbol{\beta},\mathbf{d}) - \frac{\mu}{2}(\frac{1}{n_s^2}\boldsymbol{\theta}'\mathbf{K}\boldsymbol{\theta} - \frac{2}{n_s n_t}\boldsymbol{\theta}'\mathbf{K}_{st}\mathbf{1}) \\ \text{s.t.} \quad & \mathbf{1}'(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1}) = 0, \\ & \bar{\boldsymbol{\alpha}} \le \mathbf{1}, \quad \boldsymbol{\beta} \ge \mathbf{0}, \\ & \mathbf{0} \le \boldsymbol{\alpha} \le C_2\boldsymbol{\theta}, \quad 1'\boldsymbol{\theta} = n_s, \end{aligned} \tag{37}$$

which can be solved similarly as in Algorithm 3. The only difference is that we have one more MMD regularizer in the inner optimization problem. Therefore, for MIL-CPB-PI-DA, we need solve for $(\mathbf{d}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\theta})$ at Step 4 of Algorithm 3 by optimizing the MKL problem in (37) based on the current $\mathcal{Y}$. The final classifier can be presented as $f(\mathbf{x}) = \mathbf{w}'\phi(\mathbf{x}) + b$

with $\mathbf{w} = \sum_{i=1}^{n} \alpha_i \tilde{y}_i \phi(\mathbf{x}_i)$, where α_i is the i-th entry in the final dual variable $\boldsymbol{\alpha}$, and $\tilde{y}_i = \sum_{t=1}^{r} d_t y_{t,i}$ with $y_{t,i}$ being the i-th entry of $\mathbf{y}_t$.

Similarly as sMIL-PI-DA, the primal form of (37) is also related to the formulation of SVM+, as described below,

Proposition 3 *The primal form of (37) is equivalent to the following problem,*

$$\min_{\substack{\mathbf{d}\in\mathcal{D},\mathbf{w}_t,b,\tilde{\mathbf{w}},\\ \tilde{b},\hat{\mathbf{w}},\hat{b},\boldsymbol{\eta}}} \quad P(\mathbf{d}, \mathbf{w}_t|_{t=1}^{T}, b, \tilde{\mathbf{w}}, \tilde{b}, \vec{\eta}) + \frac{\lambda}{2}\|\hat{\mathbf{w}} - \rho\mathbf{v}\|^2$$

$$+C_2 \sum_{i=1}^{n_s} \zeta(\phi(\mathbf{x}_i^s)), \tag{38}$$

$$s.t. \quad \sum_{t=1}^{T} \mathbf{w}_t' \psi_t(\mathbf{x}_i^s) \geq 1 - \xi(\tilde{\phi}(\tilde{\mathbf{x}}_i^s)) - \zeta(\phi(\mathbf{x}_i^s)),$$

$$\forall i = 1, \ldots, n^+, \tag{39}$$

$$\sum_{t=1}^{T} \mathbf{w}_t' \psi_t(\mathbf{x}_i^s) \geq 1 - \eta_i - \zeta(\phi(\mathbf{x}_i^s)),$$

$$\forall i = n^+ + 1, \ldots, n_s, \tag{40}$$

$$\xi(\tilde{\phi}(\tilde{\mathbf{x}}_i^s)) \geq 0, \quad \forall i = 1, \ldots, n^+, \tag{41}$$

$$\eta_i \geq 0, \qquad \forall i = n^+ + 1, \ldots, n_s, \tag{42}$$

$$\zeta(\phi(\mathbf{x}_i^s)) \geq 0, \quad \forall i = 1, \ldots, n_s, \tag{43}$$

where $P(\mathbf{d}, \mathbf{w}_t|_{t=1}^{T}, b, \tilde{\mathbf{w}}, \tilde{b}, \vec{\eta}) = \frac{1}{2}\sum_{t=1}^{T} \frac{\|\mathbf{w}_t\|^2}{d_t} + \frac{\gamma}{2}\|\tilde{\mathbf{w}}\|^2 + C_1 \sum_{j=1}^{n^+} \xi(\tilde{\phi}(\tilde{\mathbf{x}}_j^s)) + \sum_{j=n^++1}^{n} \eta_j$, $\zeta(\phi(\mathbf{x}_i^s)) = \hat{\mathbf{w}}'\phi(\mathbf{x}_i^s) + \hat{b}$, $\mathbf{v} = \frac{1}{n_s}\sum_{i=1}^{n_s} \phi(\mathbf{x}_i^s) - \frac{1}{n_t}\sum_{i=1}^{n_t} \phi(\mathbf{x}_i^t)$, $\lambda = \frac{(n_s C_2)^2}{\mu}$ *and* $\rho = \frac{n_s C_2}{\lambda}$, $\psi_t(\mathbf{x}_i^s)$ *is the nonlinear feature mapping of* $\mathbf{x}_i^s$ *induced by the kernel* $\mathbf{Q} \circ \mathbf{y}_t \mathbf{y}_t'$.

Again, the explanation for the regularizer $\frac{\lambda}{2}\|\hat{\mathbf{w}} - \rho\mathbf{v}\|^2$ and $\zeta(\phi(\mathbf{x}_i^s))$ is similar as those for Proposition 1. We can similarly prove Proposition 3 and the details are ignored here.

5 Experiments

In this paper, we evaluate our proposed methods for action and event recognition by using loosely labelled web videos as training data. However, it is worth mentioning that our newly proposed methods can be readily used for other applications like image retrieval and image categorization. For example, the effectiveness of our sMIL-PI and sMIL-PI-DA methods for image retrieval and image categorization has been demonstrated in our preliminary conference paper (Li et al. 2014b).

5.1 Video Event Recognition

Datasets and Features: We evaluate our proposed methods for video event recognition on the benchmark datasets Kodak (Loui et al. 2007) and CCV (Jiang et al. 2011).

We construct a new training dataset called “Flickr”, which contains the web videos crawled from Flickr by using six event names (i.e., “birthday”, “picnic”, “parade”, “show”, “sports” and “wedding”) as the queries. We remove the web videos if they are too short (i.e., the file size is smaller than 5M) or too long (i.e., the file size is larger than 100M). Finally we keep the top 300 web videos for each query as the relevant videos. For each query, we randomly sample the same number of Flickr videos that do not contain this query as one of the surrounding textual descriptions as irrelevant videos.

The Kodak dataset was used in Duan et al. (2012d) and (2012b), which contains 195 consumer videos from six event classes (i.e., “birthday”, “picnic”, “parade”, “show”, “sports” and “wedding”). The CCV dataset (Jiang et al. 2011) collected by Columbia University was also used in Duan et al. (2012b). It consists of a training set of 4659 videos and a test set of 4658 videos from 20 semantic categories. Following (Duan et al. 2012b), we only use the videos from the event related categories and we also merge “wedding ceremony”, “wedding reception” “wedding dance” as “wedding”, “non-music performance”, “music performance” as “show”, and “baseball”, “basketball”, “biking”, “ice skating”, “skiing”, “soccer”, “swimming” as “sports”. Finally, there are 2440 videos from five event classes (i.e., “birthday”, “parade”, “show”, “sports”, and “wedding”). Since different datasets have different numbers of event classes, we use the 6 (resp., 5) overlapped event classes between Flickr and Kodak (resp., CCV) for performance evaluation.

We extract both textual features and improved dense trajectory features (Wang and Schmid 2013) from the training web videos. The textual features are used as privileged information.

- **Textual feature:** A 2000-dim term-frequency (TF) feature is extracted for each video by using the top-2000 words with the highest frequency as the vocabulary. Stopword removal is performed to remove the meaningless words.
- **Visual feature:** We extract improved dense trajectory features using the source code provided in Wang and Schmid (2013). Specifically, three types of space-time (ST) features (i.e., 96-dim Histogram of Oriented Gradient, 108-dim Histogram of Optical Flow and 192-dim Motion Boundary Histogram) are used, in which we set the trajectory length as 50, the sampling stride as 16, and all the other parameters as their default values. We construct the codebook by using k-means clustering on the ST features from all videos in the training dataset to

generate 2000 clusters, and then use the bag-of-words model for each type of ST features. Finally, each video is represented as a 6000-dim feature by concatenating the 2000-dim TF feature from each type of ST feature.

As the test data does not contain textual information, we only extract improved dense trajectory features for the videos in the test set, and each test video is also represented as a 6000-dim feature.

5.1.1 Experimental Results Without Domain Adaptation

Baselines: We firstly compare our methods under the MIL-PI framework with two sets of baselines: the recent LUPI methods including pSVM+ (Vapnik and Vashist 2009) and Rank Transfer (RT) (Sharmanska et al. 2013), as well as the conventional MIL method sMIL (Bunescu and Mooney 2007). We also include SVM as a baseline, which is trained by using the visual features only. Moreover, we also compare our MIL methods with Classeme (Torresani et al. 2010) and multi-view learning methods Kernel Canonical Correlation Analysis (KCCA) and SVM-2K, because they can also be used for our application.

- *KCCA* (Hardoon et al. 2004): We apply KCCA on the training set by using the textual features and visual features, and then train the SVM classifier by using the common representations of visual features. In the testing process, the visual features of test videos are transformed into their common representations for the prediction.
- *SVM-2K* (Farquhar et al. 2005): We train the SVM-2K classifiers by using the visual features and textual features from the training samples, and apply the visual feature based classifier on the test samples for the prediction.
- *Classeme* (Torresani et al. 2010): For each word in the 2000-dim textual features, we retrieve relevant and irrelevant videos to construct positive bags and negative bags, respectively. Then we follow (Li et al. 2013) to use mi-SVM to train the classeme classifier for each word. For each training video and test video, 2000 decision values are obtained by using 2000 learnt classeme classifiers and the decision values are augmented with the visual features. Finally, we train the SVM classifiers for classifying the test videos based on the augmented features.

We also compare our MIL methods with MIML (Zhou and Zhang 2006). While we can treat the top 2000 words in the textual descriptions as noisy class labels, MIML cannot be directly applied to our task because the 2000 words may be different from the concept names. Thus, we use the decision values from the MIML classifiers as the features, similarly as in Classeme. Moreover, we additionally compare our MIL methods with the MILBoost method proposed in Leung et al. (2011) which was used for video classification.

Table 1 MAPs (%) of different methods without using domain adaptation on the Kodak and CCV datasets

Method	Test set	
	Kodak	CCV
SVM	42.84	47.16
pSVM+	44.54	48.04
RT	36.22	34.16
Classeme	43.84	46.89
MIML	42.94	47.75
MILBoost	32.77	36.63
KCCA	44.46	47.91
SVM-2K	43.69	47.78
sMIL	42.94	47.90
sMIL-PI	**46.07**	**49.13**
mi-SVM	44.23	47.68
mi-SVM-PI	**45.89**	**49.32**
MIL-CPB	44.81	47.87
MIL-CPB-PI	**46.19**	**49.21**

The results in boldface are from our methods

Experimental Settings: We train the classifiers by using the videos crawled from Flickr and evaluate the performances of different methods on the Kodak and CCV datasets, respectively. Similarly as in Li et al. (2011), we uniformly partition the 300 relevant videos crawled from Flickr into positive bags, and also randomly partition the 300 irrelevant videos into negative bags. We obtain 60 positive bags and 60 negative bags by respectively using relevant videos and irrelevant videos, in which each training bag contains five instances. The positive ratio is set as $\sigma = 0.6$, as suggested in Li et al. (2011). In our experiments, we use Gaussian kernel for visual features and linear kernel for textual features for our methods and the baseline methods except RankTransfer (RT). The objective function of RT is solved in the primal form, so we can only use linear kernel instead of Gaussian kernel for visual features.

For performance evaluation, we report the Mean Average Precision (MAP) based on all test videos. For our method, we empirically fix $C_1 = 10^2, \gamma = 10^2$ (resp., $C_1 = 10^{-2}, \gamma = 10$) for sMIL-PI (resp., mi-SVM-PI, MIL-CPB-PI). For the baseline methods, we choose the optimal parameters based on their MAPs on the test dataset.

Experimental Results: The MAPs of all methods are reported in Table 1. By additionally exploiting textual information, pSVM+, Classme, MIML, KCCA, and SVM-2K are generally better than SVM. The RT method is worse than SVM due to the use of linear kernel for visual features. The MILBoost method is also much worse than SVM, although we have carefully tuned the parameters. It is worth mentioning that pSVM+ achieves better results than Classme, MIML, RT, MILBoost and the multi-view methods (i.e., SVM-2K and KCCA) on both datasets, which demonstrates it is helpful to use textual features as privileged information.

Our MIL-PI methods generally achieve similar results. MIL-CPB-PI is the best when using Kodak as the test set. While mi-SVM-PI outperforms sMIL-PI and MIL-CPB-PI when using CCV as the test set, MIL-CPB-PI also achieves comparable results as mi-SVM-PI.

Our MIL-PI methods (i.e., sMIL-PI, mi-SVM-PI, MIL-CPB-PI) are better than pSVM+, RT, MIML, Classeme, and two existing multi-view learning methods, which demonstrates that it is beneficial to further cope with label noise of web videos as in our MIL-PI framework. Moreover, each of our MIL-PI methods also outperforms its corresponding conventional MIL method (i.e., sMIL-PI vs. sMIL, mi-SVM-PI vs. mi-SVM, MIL-CPB-PI vs. MIL-CPB), which again demonstrates it is beneficial to exploit the additional textual features from web data as privileged information.

5.1.2 Experimental Results with Domain Adaptation

Baselines: We compare our methods sMIL-PI-DA, mi-SVM-PI-DA, and MIL-CPB-PI-DA in our MIL-PI-DA framework with the existing domain adaptation methods GFK (Gong et al. 2012), SGF (Gopalan et al. 2011), SA (Fernando et al. 2013), TCA (Pan et al. 2011), KMM (Huang et al. 2007), DIP (Baktashmotlagh et al. 2013), DASVM (Bruzzone and Marconcini 2010) and STM (Chu et al. 2013). We notice that the feature-based domain adaptation methods such as GFK, SGF, SA, TCA, DIP can be combined with the SVM classifier or our MIL-PI classifiers (i.e., sMIL-PI, mi-SVM-PI, and MIL-CPB-PI), so we report two results for each domain adaptation baseline method by using the SVM classifier and the best classifier from our MIL-PI framework.
Experiment Settings: We use the same setting as in Sect. 5.1.1. In our MIL-PI-DA framework, we have two more parameters (i.e., C_2 and λ) when compared with the MIL-PI framework. Recall that $\lambda = \frac{(C_2 m)^2}{\mu}$, where m is the number of source training samples and μ is the parameter used in the dual form of our MIL-PI-DA framework. We empirically fix $C_2 = 10$ (resp., $C_2 = 10^{-5}$), $\lambda = 10^2$ for sMIL-PI-DA (resp., mi-SVM-PI-DA, MIL-CPB-PI-DA). For the baseline methods, we choose the optimal parameters based on their best MAPs on the test dataset.
Experimental Results: The MAPs of all methods are reported in Table 2.

When using the SVM classifier, some existing feature-based domain adaptation methods (SA, DIP, and SGF on Kodak as well as DIP and GFK on CCV) are worse when compared with SVM. One possible explanation is that those two-step methods may not well preserve the discriminability of features when reducing the domain distribution mismatch in the first step. For these feature-based baselines, their results after using our MIL-PI framework are better when compared with those using the SVM classifier, which again shows the effectiveness of our MIL-PI framework for video event recognition by coping with label noise and simultaneously taking advantage of the additional textual features as privileged information. However, the results of the feature-based baselines after using our MIL-PI framework are still worse than our MIL-PI-DA methods. The experimental results clearly demonstrate our domain adaptation approaches are more effective than those two-step feature-based baseline methods.

Table 2 MAPs (%) of SVM, sMIL-PI, mi-SVM-PI, MIL-CPB-PI and different domain adaptation methods on the Kodak and CCV datasets

Method	Test set	
	Kodak	CCV
SVM	42.84	47.16
sMIL-PI	46.07	49.13
sMIL-PI-DA	**47.55**	**50.32**
mi-SVM-PI	45.89	49.32
mi-SVM-PI-DA	**47.59**	**50.75**
MIL-CPB-PI	46.19	49.21
MIL-CPB-PI-DA	**49.16**	**50.66**
DASVM	45.86	47.67
STM	44.93	49.00
SA	40.30 (41.34)	47.21 (49.47)
TCA	44.24 (45.92)	48.91 (49.10)
DIP	41.56 (45.69)	44.49 (46.28)
KMM	43.94 (46.29)	48.97 (49.03)
GFK	44.19 (45.79)	45.93 (48.84)
SGF	41.19 (46.41)	47.71 (48.69)

For SA, TCA, DIP, GFK and SGF, the first number is obtained by using the SVM classifier and the second number in the parenthesis is the best result obtained by using one of our MIL-PI methods. For KMM, the first number is obtained by using the SVM classifier and the second result in the parenthesis is obtained by using our sMIL-PI method. The results in boldface are from our domain adaptation methods

Our framework is more related to KMM and STM. We also report two results for KMM because KMM can be combined with SVM or our sMIL-PI method. Particularly, the instance weights are learnt in the first step by using KMM and then we use the learnt instance weights to reweight the loss function of SVM or sMIL-PI in the second step. We observe that our sMIL-PI-DA method is better than STM and KMM when using the SVM or sMIL-PI classifier. One possible explanation is our sMIL-PI-DA method can achieve the global optimal solution by solving a convex optimization problem in one step while KMM is a two-step approach and STM can only achieve the local optimum.

We also observe that each of our methods under the domain adaptation framework MIL-PI-DA outperforms its corresponding version under the MIL-PI framework (i.e., sMIL-PI-DA vs. sMIL-PI, mi-SVM-PI-DA vs. mi-SVM-PI, MIL-CPB-PI-DA vs. MIL-CPB-PI), which shows it is helpful to reduce the domain distribution mismatch by using the MMD based regularizer. Moreover, our MIL-PI-DA methods also outperform all the existing domain adaptation baselines, which demonstrates the effectiveness of our MIL-PI-DA framework.

Finally, our newly proposed instance-level MIL-PI-DA methods (i.e., mi-SVM-PI-DA and MIL-CPB-PI-DA) achieve better results than the bag-level MIL-PI-DA method sMIL-PI-DA on both test sets, which shows it is useful to infer the instance labels in the positive bags on both datasets.

5.2 Human Action Recognition

Experimental Settings: In this section, we evaluate our MIL-PI and MIL-PI-DA framework for human action recognition on the benchmark dataset HMDB51 (Kuehne et al. 2011).

We collect a new training dataset for human action recognition by crawling short videos and their surrounding textual descriptions from YouTube website using 51 action names from the HMDB51 dataset as the queries. We use the top 200 web videos for each query as the relevant videos and randomly sample the same number of web videos that do not contain the query as one of the surrounding texts as the irrelevant videos. Then, for each action class, we construct 40 training bags, in which the size of each training bag is 5. The HMDB51 dataset contains 6766 clips from 51 action classes. As suggested in Kuehne et al. (2011), we use three testing splits as the test set, in which each split contains 30 videos for each action class.

For the YouTube dataset, we extract both the textual features and the visual features for each video. For the textual features, we extract the same 2000-dim term frequency (TF) features from the surrounding textual descriptions as in Sect. 5.1. For the visual features, we follow Wang and Schmid (2013) by utilizing Fisher vector encoding, which has shown excellent performance for human action recognition. Specifically, we adopt the improved dense trajectory features and extract four types of descriptors (i.e., 30-dim trajectory, 96-dim Histogram of Oriented Gradient, 108-dim Histogram of Optical Flow, and 192-dim Motion Boundary Histogram). Then, we generate the Fisher vector features by using 256 Gaussian Mixture Models (GMMs) for each type of descriptors, and then use PCA to reduce the dimension of the concatenated Fisher vector to 10000. As the HMDB51 dataset does not contain textual descriptions, we only extract the visual features for each video in the HMDB51 dataset.

Table 3 The accuracies (%) of different methods on the HMDB51 dataset without considering the domain distribution mismatch

Method	Accuracy
SVM	50.94
pSVM+	52.64
RT	51.42
Classeme	51.63
MIML	51.76
KCCA	51.24
SVM-2K	51.91
sMIL	51.96
sMIL-PI	**53.62**
mi-SVM	52.11
mi-SVM-PI	**53.22**
MIL-CPB	53.62
MIL-CPB-PI	**55.38**

The results in boldface are from our methods

As suggested in Kuehne et al. (2011), we evaluate the baseline methods and our methods on 3 testing splits, and report the mean accuracy over 3 splits for performance evaluation. For our MIL-PI methods and MIL-PI-DA methods, we use the same parameters as in Sect. 5.1. For the baseline methods, we choose the optimal parameters based on their mean accuracies on the test dataset. The other experimental settings are the same as in Sect. 5.1.

Experimental Results: The accuracies of all methods are reported in Tables 3 and 4. From Table 3, we observe that multi-instance learning methods sMIL, mi-SVM and MIL-CPB outperform SVM, which indicates the effectiveness of multi-instance learning methods for coping with label noise. By additionally taking advantage of textual information, pSVM+, RT, Classme, MIML, KCCA, and SVM-2K are better than SVM, and each of our MIL-PI methods is also better than its corresponding conventional MIL method (i.e., sMIL-PI vs. sMIL, mi-SVM-PI vs. mi-SVM, or MIL-CPB-PI vs. MIL-CPB).

From Table 3, we also observe that our MIL-PI methods (i.e., sMIL-PI, mi-SVM-PI, MIL-CPB-PI) are better than the baseline methods (i.e., pSVM+, RT, MIML, Classeme, and multi-view learning methods), which can additionally utilize the textual features. A possible explanation is that we additionally cope with label noise of web videos by utilizing the multi-instance learning techniques.

From Table 4, we observe that the existing domain adaptation methods DASVM, SA, DIP, KMM, GFK, and SGF are better than SVM by utilizing the unlabelled target domain samples to reduce the domain distribution mismatch. It is interesting that STM and TCA are worse than SVM, although we have carefully tuned their parameters. We also observe that our sMIL-PI-DA (resp., mi-SVM-

Table 4 The accuracies (%) of SVM, our MIL-PI methods, and different domain adaptation methods on the HMDB51 dataset. For SA, TCA, DIP, GFK and SGF, the first number is obtained by using the SVM classifier and the second number in the parenthesis is the best result obtained by using one of our MIL-PI methods. For KMM, the first number is obtained by using the SVM classifier and the second result in the parenthesis is obtained by using our sMIL-PI method.

Method	Accuracy
SVM	50.94
sMIL-PI	53.62
sMIL-PI-DA	**55.45**
mi-SVM-PI	53.22
mi-SVM-PI-DA	**57.65**
MIL-CPB-PI	55.38
MIL-CPB-PI-DA	**57.31**
DASVM	51.98
STM	37.43
SA	53.16 (55.58)
TCA	43.12 (46.95)
DIP	51.20 (55.73)
KMM	53.51 (53.77)
GFK	52.90 (54.27)
SGF	51.31 (52.77)

The results in boldface are from our methods

PI-DA, MIL-CPB-PI-DA) outperforms sMIL-PI (resp., mi-SVM-PI, MIL-CPB-PI), which shows it is beneficial to reduce the domain distribution mismatch by using our domain adaptation approach. Moreover, our MIL-PI-DA methods also outperform all the existing domain adaptation baselines.

In order to further evaluate our domain adaptation approaches, we combine the feature-based domain adaptation methods (i.e., SA, TCA, DIP, GFK, and SGF) with our MIL-PI methods (sMIL-PI, mi-SVM-PI, and MIL-CPB-PI) and combine KMM with our sMIL-PI method, similarly as discussed in Sect. 5.1. For each feature-based domain adaptation method, we report the best result obtained by using one of our three MIL-PI methods. The feature-based domain adaptation methods after using the best classifier learnt from one of our three MIL-PI methods (i.e., sMIL-PI, mi-SVM-PI, or MIL-CPB-PI) and KMM after using our sMIL-PI classifier achieve better results, because our MIL-PI methods can help handle label noise and simultaneously utilize privileged information.

Our instance-level methods mi-SVM-PI-DA and MIL-CPB-PI-DA outperform the feature-based domain adaptation methods combined with our MIL-PI methods. For SA and DIP, the results in the parenthesis are slightly better than our sMIL-PI-DA (see Table 4). However, SA and DIP are both combined with our MIL-CPB-PI method. When SA and DIP are combined with our sMIL-PI method, the result of SA and DIP are 54.01 % and 53.94 %, respectively, which are still worse than our sMIL-PI-DA method.

Table 5 MAPs (%) of our MIL-PI methods when using partial privileged information (PI) and full PI

Method	Partial PI		Full PI	
	Kodak	CCV	Kodak	CCV
sMIL-PI	**46.07**	**49.13**	45.58	48.55
mi-SVM-PI	**45.89**	**49.32**	45.41	48.38
MIL-CPB-PI	**46.19**	**49.21**	45.51	48.04

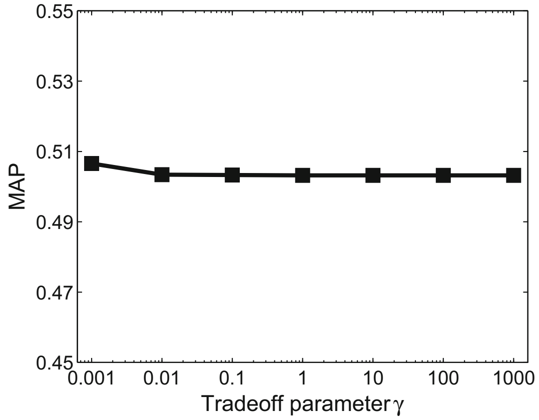

Fig. 2 MAPs of sMIL-PI-DA on the CCV dataset when using different trade-off parameter γ

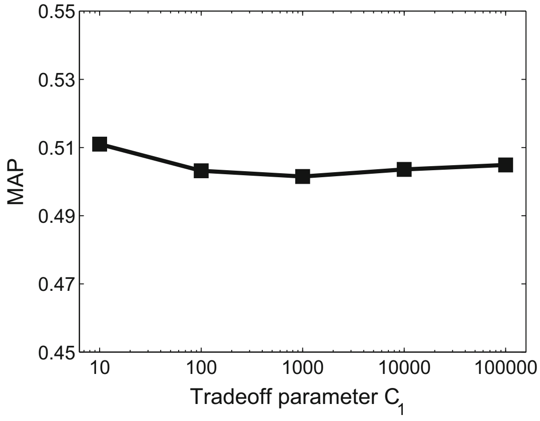

Fig. 3 MAPs of sMIL-PI-DA on the CCV dataset when using different trade-off parameter C_1

5.3 How to Utilize Privileged Information

As discussed in Sect. 3, in our MIL-PI framework, we use privileged information for relevant videos (i.e., positive bags) only, because privileged information (i.e., textual features) may not be always reliable. To verify it, we evaluate SVM+ by utilizing privileged information for all training samples. The

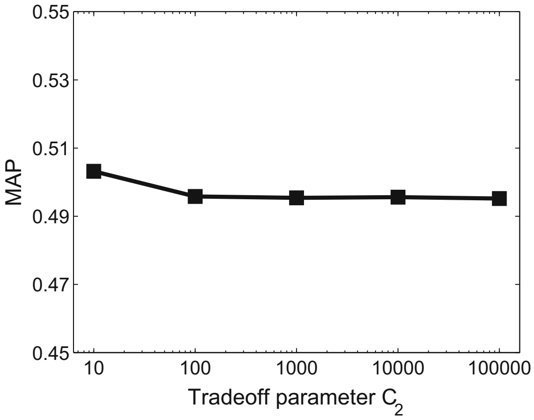

Fig. 4 MAPs of sMIL-PI-DA on the CCV dataset when using different trade-off parameter C_2, where we empirically fix $\frac{C_2}{\lambda} = 10^4$

MAPs of SVM+ are 44.08 and 47.49 % when using Kodak and CCV as the test sets, respectively, which are worse than pSVM+ on those two datasets (44.54 and 48.04 % reported in Table 1).

Similarly, we also evaluate our MIL-PI methods under two settings (i.e., full privileged information (PI) and partial privileged information (PI)). We report the results of our MIL-PI methods under two settings on the Kodak and CCV datasets in Table 5. We observe that the MAPs of our MIL-PI methods under the full PI setting are lower than their corresponding results under the partial PI setting on both datasets. These results verify our conjecture that privileged information of irrelevant web videos may not be helpful for learning robust classifiers, because the labels of irrelevant videos are generally correct while the textual features are not always reliable.

Since our MIL-PI methods with partial PI achieve better results than those with full PI, we further conjecture it may be useful to additionally learn the importance of privileged information of training samples during the training process. However, it is a non-trivial task under our setting where the labels of training samples are noisy. So we leave how to learn the importance of privileged information as our future work.

5.4 Robustness to the Parameters

Our methods are relatively robust when the trade-off parameters are set in certain ranges. Here, we study the performance variation of our sMIL-PI-DA method with respect to one parameter while fixing other parameters as their default values. Let us take the CCV dataset as an example, the MAPs of sMIL-PI-DA are in the range of [50.32 %, 50.66 %] (resp., [50.15 %, 51.10 %]) when we set $\gamma \in [10^{-3}, 10^3]$ (resp., $C_1 \in [10^1, 10^5]$), as shown in Fig 2 (resp., Fig 3). For the parameters C_2 and λ, we observe our methods are relatively robust when $\frac{C_2}{\lambda}$ is empirically fixed as 10^4. The MAPs of sMIL-PI-DA are in the range of [49.52 %, 50.32 %] when we set $C_2 \in [10^1, 10^5]$, as shown in Fig 4. We also have similar observations for our other methods and on other datasets. We will study how to decide the optimal parameters in our future work.

5.5 Comparison of Training Time

In this section, we take sMIL-PI and sMIL-PI-DA as two examples to compare the training time with the corresponding MIL method sMIL as well as other baselines. As shown in (8), our sMIL-PI method can be formulated as a quadratic programming (QP) problem with respect to two variables $\{\boldsymbol{\alpha}, \boldsymbol{\beta}\}$. Compared with sMIL, which can be formulated as a QP problem with respect to one variable $\boldsymbol{\alpha}$ only, the size of the QP problem in (8) is larger. However, it can still be efficiently solved with the existing QP solvers. Specifically, we take the CCV dataset as an example to compare the training time of sMIL-PI with other baseline methods. From Table 6, we observe that the training time of sMIL-PI is only slightly longer than sMIL, and our sMIL-PI method is much more efficient than other baseline methods.

Similarly, our sMIL-PI-DA method can also be solved as a QP problem w.r.t. three variables $\{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\theta}\}$. So it can also be efficiently solved by using the existing QP solvers. In Table 7, we take the CCV dataset as an example to compare the training time of sMIL-PI-DA with existing methods. We observe that our sMIL-PI-DA method is faster than other baseline methods except KMM. A possible expla-

Table 6 Training time of our sMIL-PI method and the baseline methods without domain adaptation on the CCV dataset

Method	SVM	pSVM+	RT	Classeme	MIML	KCCA	SVM-2K	sMIL	sMIL-PI
Time(s)	22.17	35.21	1501.51	1618.15	8785.27	88.13	96.98	18.31	21.86

Table 7 Training time of our sMIL-PI-DA method and the existing domain adaptation methods on the CCV dataset

Method	DASVM	STM	SA	TCA	DIP	KMM	GFK	SGF	sMIL-PI-DA
Time(s)	1130.05	204.74	615.79	972.95	1089.95	111.23	1932.82	3592.87	151.71

nation is that KMM solves a smaller scale QP problem w.r.t. $\boldsymbol{\theta}$ before training an SVM classifier in the second step.

6 Conclusion and Future Work

In this paper, we have proposed new MIL approaches for action and event recognition by learning from loosely labelled web data. We firstly propose a new MIL-PI framework together with three instantiations sMIL-PI, mi-SVM-PI and MIL-CPB-PI, in which we not only take advantage of the additional textual features in the training web videos but also effectively cope with noise in the loose labels of relevant training web videos. We further propose a new MIL-PI-DA framework and three instantiations sMIL-PI-DA, mi-SVM-PI-DA and MIL-CPB-PI-DA, which can additionally reduce the data distribution mismatch between the training and test videos. By using freely available web videos as training data, our approaches are inherently not limited by any predefined lexicon. Extensive experiments clearly demonstrate our proposed approaches are effective for action and event recognition. In future work, we will study how to automatically decide the optimal trade-off parameters for our methods. We will also investigate how to learn the importance of privileged information.

Acknowledgments This research was supported by funding from the Faculty of Engineering & Information Technologies, The University of Sydney, under the Faculty Research Cluster Program. This work was also supported by the Singapore MoE Tier 2 Grant (ARC42/13).

Appendix 1: Detailed Derivations for (16)

We provide the complete derivations for (16). For ease of presentation, we define

$$F(\hat{\boldsymbol{\alpha}}, \boldsymbol{\beta}) = \frac{1}{2\gamma}(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1})'\tilde{\mathbf{K}}(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1}).$$

Then, the problem in (15) can be rewritten as,

$$\min_{\mathbf{d}} \max_{(\boldsymbol{\alpha},\boldsymbol{\beta})\in\mathcal{S}} \quad \mathbf{1}'\boldsymbol{\alpha} - \frac{1}{2}\boldsymbol{\alpha}'\left(\sum_{t=1}^{T} d_t \mathbf{Q} \circ \mathbf{y}_t\mathbf{y}_t'\right)\boldsymbol{\alpha} - F(\hat{\boldsymbol{\alpha}}, \boldsymbol{\beta})$$
$$\text{s.t.} \quad \sum_{t=1}^{T} d_t = 1, \quad d_t \geq 0, \quad \forall t = 1, \ldots, T \tag{44}$$

Let us introduce a dual variable τ for the constraint $\sum_{t=1}^{T} d_t = 1$ and another dual variable ν_t for each constraint $d_t \geq 0$ in problem (44), we arrive at its Lagrangian as follows,

$$\mathcal{L} = \mathbf{1}'\boldsymbol{\alpha} - \frac{1}{2}\boldsymbol{\alpha}'\left(\sum_{t=1}^{T} d_t \mathbf{Q} \circ \mathbf{y}_t\mathbf{y}_t'\right)\boldsymbol{\alpha} - F(\hat{\boldsymbol{\alpha}}, \boldsymbol{\beta})$$
$$+ \tau\left(\sum_{t=1}^{T} d_t - 1\right) - \sum_{t=1}^{T} \nu_t d_t. \tag{45}$$

The derivative of the Lagrangian w.r.t. d_t can be written as,

$$\frac{\partial\mathcal{L}}{\partial d_t} = -\frac{1}{2}\boldsymbol{\alpha}'\left(\mathbf{Q} \circ \mathbf{y}_t\mathbf{y}_t'\right)\boldsymbol{\alpha} + \tau - \nu_t, \quad \forall t = 1, \ldots, T.$$

Let us set $\frac{\partial\mathcal{L}}{\partial d_t} = 0$ and consider $\nu_t \geq 0$, we have

$$\frac{1}{2}\boldsymbol{\alpha}'\left(\mathbf{Q} \circ \mathbf{y}_t\mathbf{y}_t'\right)\boldsymbol{\alpha} \leq \tau, \quad \forall t = 1, \ldots, T.$$

By substituting $\frac{\partial\mathcal{L}}{\partial d_t} = 0$ into the Lagrangian, we obtain the duality of (44) as follows,

$$\max_{\tau} \max_{(\boldsymbol{\alpha},\boldsymbol{\beta})\in\mathcal{S}} \quad \mathbf{1}'\boldsymbol{\alpha} - F(\hat{\boldsymbol{\alpha}}, \boldsymbol{\beta}) - \tau$$
$$\text{s.t.} \quad \frac{1}{2}\boldsymbol{\alpha}'\left(\mathbf{Q} \circ \mathbf{y}_t\mathbf{y}_t'\right)\boldsymbol{\alpha} \leq \tau, \quad \forall t = 1, \ldots, T, \tag{46}$$

which is the same as (16). We complete the derivations here.

Appendix 2: Solution to the MKL Problem at Step 4 of Algorithm 3

At Step 4 of Algorithm 3, we solve an MKL problem in (15) by setting $\mathcal{Y} = \mathcal{C}$. As $\mathcal{C}$ contains only a small number of label vectors, so the number of base kernels is not large. Now we give the algorithm for solving the MKL problem in (15) with a few kernels.

Let us denote $\tilde{T} = |\mathcal{C}|$ as the number of label vectors in $\mathcal{C}$. We also define $\mathbf{d} = [d_1, \ldots, d_{\tilde{T}}]'$ as the vector of kernel coefficients, and $\mathcal{D} = \{\mathbf{d}'\mathbf{1} = 1, \mathbf{d} \geq 0\}$. Note we use the same symbols $\mathbf{d}$ and $\mathcal{D}$ as in (15) for simplicity, but the dimensionality of $\mathbf{d}$ (i.e., $\tilde{T} = |\mathcal{C}|$) is much smaller than that in (15) (i.e., $T = |\mathcal{Y}|$). Now, we write the primal form of (15) as follows,

$$\min_{\substack{\mathbf{d}\in\mathcal{D} \\ \tilde{\mathbf{w}}, \tilde{b}, \mathbf{w}_t, \boldsymbol{\eta}}} \quad \frac{1}{2}\sum_{t=1}^{\tilde{T}} \frac{\|\mathbf{w}_t\|^2}{d_t} + \frac{\gamma}{2}\|\tilde{\mathbf{w}}\|^2$$
$$+ C_1 \sum_{i=1}^{n^+} \xi(\tilde{\phi}(\tilde{\mathbf{x}}_i)) + \sum_{i=n^+ +1}^{n} \eta_i, \tag{47}$$
$$\text{s.t.} \quad \sum_{t=1}^{\tilde{T}} \mathbf{w}_t'\psi_t(\mathbf{x}_i) \geq 1 - \xi(\tilde{\phi}(\tilde{\mathbf{x}}_i)),$$
$$\xi(\tilde{\phi}(\tilde{\mathbf{x}}_i)) \geq 0, \qquad i = 1, \ldots, n^+,$$

$$\sum_{t=1}^{\tilde{T}} \mathbf{w}_t' \psi_t(\mathbf{x}_i) \geq 1 - \eta_i,$$
$$\eta_i \geq 0, \qquad i = n^+ + 1, \dots, n, \tag{48}$$

where $\psi_t(\mathbf{x}_i)$ is the nonlinear feature of $\mathbf{x}_i$ induced by the kernel $\mathbf{Q} \circ \mathbf{y}_t\mathbf{y}_t'$, and $\mathbf{w}_t = d_t \sum_{i=1}^{n} \alpha_i y_i^t \phi(\mathbf{x}_i)$ with y_i^t being the i-the entry in $\mathbf{y}_t$.

The above problem is a convex problem w.r.t. $\tilde{\mathbf{w}}, \tilde{b}, \mathbf{w}_t, \boldsymbol{\eta}$, and $\mathbf{d}$, so we can achieve the global optimum by alternatively optimizing two set of variables $\{\tilde{\mathbf{w}}, \tilde{b}, \mathbf{w}_t, \boldsymbol{\eta}\}$, and $\mathbf{d}$.

Fix d: When $\mathbf{d}$ is fixed, we solve for $\{\tilde{\mathbf{w}}, \tilde{b}, \mathbf{w}_t, \boldsymbol{\eta}\}$ by optimizing the dual problem in (15), i.e.,

$$\max_{(\boldsymbol{\alpha},\boldsymbol{\beta})\in\mathcal{S}} \quad \mathbf{1}'\boldsymbol{\alpha} - \frac{1}{2}\boldsymbol{\alpha}' \left(\sum_{t=1}^{\tilde{T}} d_t \mathbf{Q} \circ \mathbf{y}_t\mathbf{y}_t' \right) \boldsymbol{\alpha}$$
$$- \frac{1}{2\gamma}(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1})'\tilde{\mathbf{K}}(\hat{\boldsymbol{\alpha}} + \boldsymbol{\beta} - C_1\mathbf{1}), \tag{49}$$

which is a quadratic programming problem w.r.t. $(\boldsymbol{\alpha}, \boldsymbol{\beta})$, and can be solved by using any QP solver.

Fix $\{\tilde{\mathbf{w}}, \tilde{b}, \mathbf{w}_t, \boldsymbol{\eta}\}$: The optimization problem w.r.t. $\mathbf{d}$ can be written as,

$$\min_{\mathbf{d}} \quad \frac{1}{2} \sum_{t=1}^{\tilde{T}} \frac{\|\mathbf{w}_t\|^2}{d_t}$$
$$\text{s.t.} \quad \mathbf{d}'\mathbf{1} = 1, \quad \mathbf{d} \geq 0, \tag{50}$$

which is the same as solving the kernel coefficients in ℓ_p-norm MKL (Kloft et al. 2011) when $p = 1$, and has a closed-form solution as below,

$$d_t = \frac{\|\mathbf{w}_t\|}{\sum_{t=1}^{\tilde{T}} \|\mathbf{w}_t\|}, \tag{51}$$

where $\|\mathbf{w}_t\|$ can be calculated from $\|\mathbf{w}_t\|^2 = d_t^2 \boldsymbol{\alpha}'(\mathbf{Q} \circ \mathbf{y}_t\mathbf{y}_t')\boldsymbol{\alpha}$. We repeat above two steps until the objective value of (49) converges.

Appendix 3: Proof of Proposition 1

Proof By introducing the dual variables $\hat{\boldsymbol{\alpha}} = [\alpha_1, \dots, \alpha_{L^+}]' \in \mathbb{R}^{L^+}$ for the constraints in (23), $\bar{\boldsymbol{\alpha}} = [\alpha_{L^+ +1}, \dots, \alpha_m]' \in \mathbb{R}^{m-L^+}$ for the constraints (24), $\hat{\boldsymbol{\beta}} = [\beta_1, \dots, \beta_{L^+}]' \in \mathbb{R}^{L^+}$ for the constraints in (25), $\bar{\boldsymbol{\beta}} = [\beta_{L^+ +1}, \dots, \beta_m]' \in \mathbb{R}^{m-L^+}$ for the constraints in (26), and $\boldsymbol{\nu} = [\nu_1, \dots, \nu_m]'$ for the constraints in (27), we arrive at its Lagrangian as follows:

$$\mathcal{L} = \frac{1}{2}\left(\|\mathbf{w}\|^2 + \gamma\|\tilde{\mathbf{w}}\|^2\right) + C_1 \sum_{i=1}^{L^+} (\tilde{\mathbf{w}}'\tilde{\mathbf{z}}_i^s + \tilde{b})$$
$$+ \sum_{i=L^+ +1}^{m} \eta_i + \frac{\lambda}{2}\|\hat{\mathbf{w}} - \rho\mathbf{v}\|^2 + C_2 \sum_{i=1}^{m} (\hat{\mathbf{w}}'\mathbf{z}_i^s + \hat{b})$$
$$- \sum_{i=1}^{L^+} \hat{\alpha}_i (\mathbf{w}'\mathbf{z}_i^s + b - p_i + \tilde{\mathbf{w}}'\tilde{\mathbf{z}}_i^s + \tilde{b} + \hat{\mathbf{w}}'\mathbf{z}_i^s + \hat{b})$$
$$- \sum_{i=L^+ +1}^{m} \bar{\alpha}_i (-\mathbf{w}'\mathbf{z}_i^s - b - 1 + \eta_i + \hat{\mathbf{w}}'\mathbf{z}_i^s + \hat{b})$$
$$- \sum_{i=1}^{L^+} \hat{\beta}_i (\tilde{\mathbf{w}}'\tilde{\mathbf{z}}_i^s + \tilde{b}) - \sum_{i=L^+ +1}^{m} \bar{\beta}_i \eta_i - \sum_{i=1}^{m} \nu_i (\hat{\mathbf{w}}'\mathbf{z}_i^s + \hat{b}), \tag{52}$$

Let us define $\boldsymbol{\alpha} = [\hat{\boldsymbol{\alpha}}', \bar{\boldsymbol{\alpha}}']'$, $\boldsymbol{\beta} = [\hat{\boldsymbol{\beta}}', \bar{\boldsymbol{\beta}}']'$, $\mathbf{Z} = [\mathbf{z}_1^s, \dots, \mathbf{z}_m^s]$, $\tilde{\mathbf{Z}} = [\tilde{\mathbf{z}}_1^s, \dots, \tilde{\mathbf{z}}_{L^+}^s]$, and $\mathbf{y} = [\mathbf{1}_{L^+}', -\mathbf{1}_{m-L^+}']'$, then the derivatives of the Lagrangian w.r.t. $\mathbf{w}, b, \tilde{\mathbf{w}}, \tilde{b}, \hat{\mathbf{w}}, \hat{b}, \boldsymbol{\eta}$ can be obtained as follows:

$$\frac{\partial\mathcal{L}}{\partial\mathbf{w}} = \mathbf{w} - \mathbf{Z}(\boldsymbol{\alpha} \circ \mathbf{y}),$$
$$\frac{\partial\mathcal{L}}{\partial b} = -\boldsymbol{\alpha}'\mathbf{y},$$
$$\frac{\partial\mathcal{L}}{\partial\tilde{\mathbf{w}}} = \gamma\tilde{\mathbf{w}} - \tilde{\mathbf{Z}}(\hat{\boldsymbol{\alpha}} + \hat{\boldsymbol{\beta}} - C_1\mathbf{1}_{L^+}),$$
$$\frac{\partial\mathcal{L}}{\partial\tilde{b}} = -\mathbf{1}_{L^+}'(\hat{\boldsymbol{\alpha}} + \hat{\boldsymbol{\beta}} - C_1\mathbf{1}_{L^+}),$$
$$\frac{\partial\mathcal{L}}{\partial\hat{\mathbf{w}}} = \lambda\hat{\mathbf{w}} - \lambda\rho\mathbf{v} - \mathbf{Z}(\boldsymbol{\alpha} + \boldsymbol{\nu} - C_2\mathbf{1}_m),$$
$$\frac{\partial\mathcal{L}}{\partial\hat{b}} = -\mathbf{1}_m'(\boldsymbol{\alpha} + \boldsymbol{\nu} - C_2\mathbf{1}),$$
$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{\eta}} = \mathbf{1}_{m-L^+} - \bar{\boldsymbol{\alpha}} - \bar{\boldsymbol{\beta}}.$$

By setting those derivatives to zeros, we have the following equations:

$$\mathbf{w} = \mathbf{Z}(\boldsymbol{\alpha} \circ \mathbf{y}), \tag{53}$$
$$\tilde{\mathbf{w}} = \frac{1}{\gamma}\tilde{\mathbf{Z}}(\hat{\boldsymbol{\alpha}} + \hat{\boldsymbol{\beta}} - C_1\mathbf{1}_{L^+}), \tag{54}$$
$$\hat{\mathbf{w}} = \rho\mathbf{v} + \frac{1}{\lambda}\mathbf{Z}(\boldsymbol{\alpha} + \boldsymbol{\nu} - C_2\mathbf{1}_m), \tag{55}$$

as well as the following constraints, $\boldsymbol{\alpha}'\mathbf{y} = 0$, $\mathbf{1}_{L^+}'(\hat{\boldsymbol{\alpha}} + \hat{\boldsymbol{\beta}} - C_1\mathbf{1}_{L^+}) = 0$, $\mathbf{1}_m'(\boldsymbol{\alpha} + \boldsymbol{\nu} - C_2\mathbf{1}_m) = 0$, $\bar{\boldsymbol{\alpha}} \leq \mathbf{1}_{m-L^+}$. Substituting the equations (53), (54) and (55) into (52) and considering $\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\nu} \geq \mathbf{0}$, we obtain the following dual form,

$$\min_{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\nu}} \quad -\mathbf{p}'\boldsymbol{\alpha} + \frac{1}{2}\boldsymbol{\alpha}'(\mathbf{K} \circ \mathbf{y}\mathbf{y}')\boldsymbol{\alpha}$$
$$+ \frac{1}{2\gamma}(\hat{\boldsymbol{\alpha}} + \hat{\boldsymbol{\beta}} - C_1\mathbf{1})'\tilde{\mathbf{K}}(\hat{\boldsymbol{\alpha}} + \hat{\boldsymbol{\beta}} - C_1\mathbf{1})$$
$$+ \frac{1}{2\lambda}(\boldsymbol{\alpha} + \boldsymbol{\nu} - C_2\mathbf{1}_m)'\mathbf{K}(\boldsymbol{\alpha} + \boldsymbol{\nu} - C_2\mathbf{1}_m)$$

$$+\rho\mathbf{v}'\mathbf{Z}(\boldsymbol{\alpha}+\boldsymbol{\nu}-C_2\mathbf{1}_m) \tag{56}$$

$$\begin{aligned}\text{s.t.}\quad & \boldsymbol{\alpha}'\mathbf{y}=0,\quad \mathbf{1}'_{L^+}(\hat{\boldsymbol{\alpha}}+\hat{\boldsymbol{\beta}}-C_1\mathbf{1}_{L^+})=0,\\ & \bar{\boldsymbol{\alpha}}\leq\mathbf{1}_{m-L^+},\\ & \mathbf{1}'_m(\boldsymbol{\alpha}+\boldsymbol{\nu}-C_2\mathbf{1}_m)=0,\quad \boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\nu}\geq\mathbf{0},\end{aligned} \tag{57}$$

Let us define $\boldsymbol{\theta}=\frac{1}{C_2}(\boldsymbol{\alpha}+\boldsymbol{\nu})$, then the constraint $\mathbf{1}'_m(\boldsymbol{\alpha}+\boldsymbol{\nu}-C_2\mathbf{1}_m)=0$ becomes $\mathbf{1}'_m\boldsymbol{\theta}=m$, and the constraint $\boldsymbol{\nu}\geq\mathbf{0}$ becomes $\boldsymbol{\alpha}\leq C_2\boldsymbol{\theta}$. Let us define the feasible set for $(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\nu})$ as $\mathcal{A}=\{\boldsymbol{\alpha}'\mathbf{y}=0,\mathbf{1}'_{L^+}(\hat{\boldsymbol{\alpha}}+\hat{\boldsymbol{\beta}}-C_1\mathbf{1}_{L^+})=0,\bar{\boldsymbol{\alpha}}\leq\mathbf{1}_{m-L^+},\mathbf{1}'_m\boldsymbol{\theta}=m,\boldsymbol{\alpha}\leq C_2\boldsymbol{\theta},\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\theta}\geq\mathbf{0}\}$. Substituting $\boldsymbol{\theta}=\frac{1}{C_2}(\boldsymbol{\alpha}+\boldsymbol{\nu})$ into (56), we arrive at,

$$\begin{aligned}\min_{(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\theta})\in\mathcal{A}}\quad & -\mathbf{p}'\boldsymbol{\alpha}+\frac{1}{2}\boldsymbol{\alpha}'(\mathbf{K}\circ\mathbf{y}\mathbf{y}')\boldsymbol{\alpha}\\ & +\frac{1}{2\gamma}(\hat{\boldsymbol{\alpha}}+\hat{\boldsymbol{\beta}}-C_1\mathbf{1})'\tilde{\mathbf{K}}(\hat{\boldsymbol{\alpha}}+\hat{\boldsymbol{\beta}}-C_1\mathbf{1})\\ & +\frac{(C_2)^2}{2\lambda}(\boldsymbol{\theta}-\mathbf{1}_m)'\mathbf{K}(\boldsymbol{\theta}-\mathbf{1}_m)+\rho C_2\mathbf{v}'\mathbf{Z}(\boldsymbol{\theta}-\mathbf{1}_m)\end{aligned} \tag{58}$$

Recall in the main text we have defined $H(\boldsymbol{\alpha},\boldsymbol{\beta})=-\mathbf{p}'\boldsymbol{\alpha}+\frac{1}{2}\boldsymbol{\alpha}'(\mathbf{K}\circ\mathbf{y}\mathbf{y}')\boldsymbol{\alpha}+\frac{1}{2\gamma}(\hat{\boldsymbol{\alpha}}+\hat{\boldsymbol{\beta}}-C_1\mathbf{1})'\tilde{\mathbf{K}}(\hat{\boldsymbol{\alpha}}+\hat{\boldsymbol{\beta}}-C_1\mathbf{1})$, then we simplify the objective function in (58) as follows,

$$\begin{aligned}\min_{(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\theta})\in\mathcal{A}}\quad & H(\boldsymbol{\alpha},\boldsymbol{\beta})+\frac{(C_2)^2}{2\lambda}(\boldsymbol{\theta}-\mathbf{1}_m)'\mathbf{K}(\boldsymbol{\theta}-\mathbf{1}_m)\\ & +\rho C_2\mathbf{v}'\mathbf{Z}(\boldsymbol{\theta}-\mathbf{1}_m)\end{aligned} \tag{59}$$

Now, we derive the objective function as follows,

$$\begin{aligned}\min_{(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\theta})\in\mathcal{A}}\quad & H(\boldsymbol{\alpha},\boldsymbol{\beta})+\frac{(C_2)^2}{2\lambda}(\boldsymbol{\theta}-\mathbf{1}_m)'\mathbf{K}(\boldsymbol{\theta}-\mathbf{1}_m)\\ & +\rho C_2\mathbf{v}'\mathbf{Z}(\boldsymbol{\theta}-\mathbf{1}_m)\end{aligned} \tag{60}$$

$$\begin{aligned}\Leftrightarrow\min_{(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\theta})\in\mathcal{A}}\quad & H(\boldsymbol{\alpha},\boldsymbol{\beta})+\frac{(C_2)^2}{2\lambda}(\boldsymbol{\theta}'\mathbf{K}\boldsymbol{\theta}-2\mathbf{1}'_m\mathbf{K}\boldsymbol{\theta})\\ & +\rho C_2\mathbf{v}'\mathbf{Z}\boldsymbol{\theta}\end{aligned} \tag{61}$$

$$\begin{aligned}\Leftrightarrow\min_{(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\theta})\in\mathcal{A}}\quad & H(\boldsymbol{\alpha},\boldsymbol{\beta})+\frac{(C_2)^2}{2\lambda}\boldsymbol{\theta}'\mathbf{K}\boldsymbol{\theta}-\frac{(C_2)^2}{\lambda}\mathbf{1}'_m\mathbf{K}\boldsymbol{\theta}\\ & +\frac{\rho C_2}{m}\mathbf{1}'_m\mathbf{K}\boldsymbol{\theta}-\frac{\rho C_2}{n_t}\mathbf{1}'_{n_t}\mathbf{K}_{ts}\boldsymbol{\theta}\end{aligned} \tag{62}$$

where in (61) we omit the constant terms, and in (62) we use the equation that $\mathbf{v}'\mathbf{Z}=\frac{1}{m}\mathbf{1}'_m\mathbf{K}-\frac{1}{n_t}\mathbf{1}'_{n_t}\mathbf{K}_{ts}$ with $\mathbf{K}_{ts}\in\mathbb{R}^{n_t\times m}$ being the kernel matrix between the target domain samples and the source domain samples. Let us define $\lambda=\frac{(C_2m)^2}{\mu}$ and $\rho=\frac{C_2m}{\lambda}=\frac{\mu}{C_2m}$ and omit the constant term, then the problem in (62) becomes

$$\begin{aligned}\min_{(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\theta})\in\mathcal{A}}\quad & H(\boldsymbol{\alpha},\boldsymbol{\beta})+\frac{\mu}{2m^2}\boldsymbol{\theta}'\mathbf{K}\boldsymbol{\theta}-\frac{\mu}{mn_t}\mathbf{1}'_{n_t}\mathbf{K}_{ts}\boldsymbol{\theta}\\ \Leftrightarrow\min_{(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\theta})\in\mathcal{A}}\quad & H(\boldsymbol{\alpha},\boldsymbol{\beta})+\frac{\mu}{2m^2}\boldsymbol{\theta}'\mathbf{K}\boldsymbol{\theta}-\frac{\mu}{mn_t}\mathbf{1}'_{n_t}\mathbf{K}_{ts}\boldsymbol{\theta}\\ & +\frac{\mu}{2n_t^2}\mathbf{1}'_{n_t}\mathbf{K}_t\mathbf{1}_{n_t}\end{aligned} \tag{63}$$

$$\Leftrightarrow\min_{(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\theta})\in\mathcal{A}}\quad H(\boldsymbol{\alpha},\boldsymbol{\beta})+\frac{\mu}{2}\|\frac{1}{m}\sum_{i=1}^{m}\theta_i\mathbf{z}_i^s-\frac{1}{n_t}\sum_{i=1}^{n_t}\mathbf{z}_i^t\|^2, \tag{64}$$

where in (63) we add a constant $\frac{\mu}{2n_t^2}\mathbf{1}'_{n_t}\mathbf{K}_t\mathbf{1}_{n_t}$ to the objective function with $\mathbf{K}_t\in\mathbb{R}^{n_t\times n_t}$ being the kernel matrix on the target domain samples. Note the problem in (64) is exactly the problem in (18). We complete the proof here. □

References

Aggarwal, J. K., & Ryoo, M. S. (2011). Human activity analysis: A review. *ACM Computing Surveys (CSUR)*, *43*(3), 16.

Andrews, S., Tsochantaridis, I., & Hofmann, T. (2003). Support vector machines for multiple-instance learning. In *Advances in Neural Information Processing Systems (NIPS)* (pp. 561–568).

Baktashmotlagh, M., Harandi, M., & Brian Lovell, M. S. (2013). Unsupervised domain adaptation by domain invariant projection. In *IEEE International Conference on Computer Vision (ICCV)* (pp. 769–776).

Bergamo, A., & Torresani, L. (2010). Exploiting weakly-labeled web images to improve object classification: a domain adaptation approach. In *Advances in Neural Information Processing Systems (NIPS)* (pp. 181–189).

Bobick, A. F. (1997). Movement, activity and action: The role of knowledge in the perception of motion. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *352*(1358), 1257–1265.

Bootkrajang, J., & Kabán, A. (2014). Learning kernel logistic regression in the presence of class label noise. *Pattern Recognition*, *47*(11), 3641–3655.

Bruzzone, L., & Marconcini, M. (2010). Domain adaptation problems: A DASVM classification technique and a circular validation strategy. *T-PAMI*, *32*(5), 770–787.

Bunescu, R. C., & Mooney, R. J. (2007). Multiple instance learning for sparse positive bags. In *International Conference on Machine learning (ICML)* (pp. 105–112).

Chang, S. F., Ellis, D., Jiang, W., Lee, K., Yanagawa, A., Loui, A. C., & Luo, J. (2007). Large-scale multimodal semantic concept detection for consumer video. In *International Workshop on Multimedia Information Retrieval* (pp. 255–264).

Chen, L., Duan, L., & Xu, D. (2013a) Event recognition in videos by learning from heterogeneous web sources. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 2666–2673).

Chen, X., Shrivastava, A., & Gupta, A. (2013b) NEIL: Extracting visual knowledge from web data. In *IEEE International Conference on Computer Vision (ICCV)* (pp. 1409–1416).

Chen, Y., Bi, J., & Wang, J. Z. (2006). MILES: Multiple-instance learning via embedded instance selection. *T-PAMI*, *28*(12), 1931–1947.

Chu, W. S., DelaTorre, F., & Cohn, J. (2013) Selective transfer machine for personalized facial action unit detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3515–3522).

Duan, L., Li, W., Tsang, I. W., & Xu, D. (2011). Improving web image search by bag-based re-ranking. *T-IP*, *20*(11), 3280–3290.

Duan, L., Tsang, I. W., & Xu, D. (2012a). Domain transfer multiple kernel learning. *T-PAMI*, *34*(3), 465–479.

Duan, L., Xu, D., & Chang, S. F. (2012b). Exploiting web images for event recognition in consumer videos: A multiple source domain adaptation approach. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1338–1345).

Duan, L., Xu, D., & Tsang, I. W. (2012c). Domain adaptation from multiple sources: A domain-dependent regularization approach. *T-NNLS*, *23*(3), 504–518.

Duan, L., Xu, D., Tsang, I. W., & Luo, J. (2012d). Visual event recognition in videos by learning from web data. *T-PAMI*, *34*(9), 1667–1680.

Farhadi, A., Endres, I., Hoiem, D., & Forsyth, D. (2009). Describing objects by their attributes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1778–1785).

Farquhar, J. D. R., Hardoon, D. R., Meng, H., Shawe-Taylor, J., & Szedmak, S. (2005). Two view learning: SVM-2K, theory and practice. In *NIPS*.

Fergus, R., Fei-Fei, L., Perona, P., & Zisserman, A. (2005). Learning object categories from Google's image search. In *ICCV*.

Fernando, B., Habrard, A., Sebban, M., & Tuytelaars, T. (2013). Unsupervised visual domain adaptation using subspace alignment. In *ICCV*.

Ferrari, V., & Zisserman, A. (2007). Learning visual attributes. In *Advances in Neural Information Processing Systems (NIPS)* (pp. 433–440).

Fouad, S., Tino, P., Raychaudhury, S., & Schneider, P. (2013). Incorporating privileged information through metric learning. *T-NNLS*, *24*(7), 1086–1098.

Gehler, P. V., & Nowozin, S. (2008). Infinite kernel learning.Tech. rep., Max Planck Institute for Biological Cybernetics. In *NIPS Workshop on Kernel Learning: Automatic Selection of Optimal Kernels*.

Gong, B., Shi, Y., Sha, F., & Grauman, K. (2012). Geodesic flow kernel for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 2066–2073).

Gopalan, R., Li, R., & Chellappa, R. (2011). Domain adaptation for object recognition: An unsupervised approach. In *IEEE International Conference on Computer Vision (ICCV)* (pp. 999–1006).

Gretton, A., Rasch, K. M., Schlkopf, B., & Smola, A. (2012). A kernel two-sample test. *JMLR*, *13*, 723–773.

Hardoon, D. R., Szedmak, S., & Shawe-taylor, J. (2004). Canonical correlation analysis: An overview with application to learning methods. *Neural Computation*, *16*(12), 2639–2664.

Hu, Y., Cao, L., Lv, F., Yan, S., Gong, Y., & Huang, T. S. (2009). Action detection in complex scenes with spatial and temporal ambiguities. In *IEEE International Conference on Computer Vision (ICCV)* (pp. 128–135).

Huang, J., Smola, A., Gretton, A., Borgwardt, K., & Scholkopf, B. (2007). Correcting sample selection bias by unlabeled data. In *Advances in Neural Information Processing Systems (NIPS)* (pp. 601–608).

Hwang, S. J., & Grauman, K. (2012). Learning the relative importance of objects from tagged images for retrieval and cross-modal search. *IJCV*, *100*(2), 134–153.

Jiang, Y. G., Ye, G., Chang, S. F., Ellis, D., & Loui, A. C. (2011). Consumer video understanding: A benchmark database and an evaluation of human and machine performance. In *International Conference on Multimedia Retrieval (ICMR)* (p. 29).

Jiang, Y. G., Bhattacharya, S., Chang, S. F., & Shah, M. (2013). High-level event recognition in unconstrained videos. *International Journal of Multimedia Information Retrieval*, *2*(2), 73–101.

Kloft, M., Brefeld, U., Sonnenburg, S., & Zien, A. (2011). ℓ_p-norm multiple kernel learning. *JMLR*, *12*, 953–997.

Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., & Serre, T. (2011). HMDB: a large video database for human motion recognition. In *IEEE International Conference on Computer Vision (ICCV)* (pp. 2556–2563).

Kulis, B., Saenko, K., & Darrell, T. (2011). What you saw is not what you get: Domain adaptation using asymmetric kernel transforms. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1785–1792).

Le, Q. V., Zou, W. Y., Yeung, S. Y., & Ng, A.Y. (2011). Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3361–3368).

Leung, T., Song, Y., & Zhang, J. (2011). Handling label noise in video classification via multiple instance learning. In *IEEE International Conference on Computer Vision (ICCV)* (pp. 2056–2063).

Li, Q., Wu, J., & Tu, Z. (2013). Harvesting mid-level visual concepts from large-scale Internet images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 851–858).

Li, W., Duan, L., Xu, D., & Tsang, I. W. (2011). Text-based image retrieval using progressive multi-instance learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 2368–2375).

Li, W., Duan, L., Tsang, I.W., & Xu, D. (2012a). Batch mode adaptive multiple instance learning for computer vision tasks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 2368–2375).

Li, W., Duan, L., Tsang, I.W., & Xu, D. (2012b). Co-labeling: A new multi-view learning approach for ambiguous problems. In *IEEE International Conference on Data Mining (ICDM)* (pp. 419–428).

Li, W., Duan, L., Xu, D., & Tsang, I. W. (2014a). Learning with augmented features for supervised and semi-supervised heterogeneous domain adaptation. *T-PAMI*, *36*(6), 1134–1148.

Li, W., Niu, L., & Xu, D. (2014b). Exploiting privileged information from web data for image categorization. In *European Conference on Computer Vision (ECCV)* (pp. 437–452).

Li, Y.-F., Tsang, I. W., Kwok, J. T., & Zhou, Z.-H. (2009). Tighter and convex maximum margin clustering. In *International Conference on Artificial Intelligence and Statistics* (pp. 344–351).

Liang, L., Cai, F., & Cherkassky, V. (2009). Predictive learning with structured (grouped) data. *Neural Networks*, *22*, 766–773.

Lin, Z., Jiang, Z., & Davis, L. S. (2009). Recognizing actions by shape-motion prototype trees. In *IEEE International Conference on Computer Vision (ICCV)* (pp. 444–451).

Loui, A., Luo, J., Chang, S. F., Ellis, D., Jiang, W., Kennedy, L., Lee, K., & Yanagawa, A. (2007). Kodak's consumer video benchmark data set: concept definition and annotation. In *International Workshop on Multimedia Information Retrieval* (pp. 245–254).

Morariu, V.I., & Davis, L.S. (2011). Multi-agent event recognition in structured scenarios. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3289–3296).

Natarajan, N., Dhillon, I. S., Ravikumar, P. K., & Tewari, A. (2013). Learning with noisy labels. In *Advances in Neural Information Processing Systems*, pp 1196–1204.

Pan, S. J., Tsang, I. W., Kwok, J. T., & Yang, Q. (2011). Domain adaptation via transfer component analysis. *T-NN*, *22*(2), 199–210.

Schroff, F., Criminisi, A., & Zisserman, A. (2011). Harvesting image databases from the web. *T-PAMI*, *33*(4), 754–766.

Sharmanska, V., Quadrianto, N., Lampert, C. H. (2013). Learning to rank using privileged information. In *IEEE International Conference on Computer Vision (ICCV)* (pp. 825–832).

Shi, Y., Huang, Y., Minnen, D., Bobick, A., & Essa, I. (2004). Propagation networks for recognition of partially ordered sequential action. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (vol. 2, pp. II-862–II-869).

Torralba, A., & Efros, A.A. (2011). Unbiased look at dataset bias. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1521–1528).

Torralba, A., Fergus, R., & Freeman, W. T. (2008). 80 million tiny images: A large data set for nonparametric object and scene recognition. *T-PAMI*, *30*(11), 1958–1970.

Torresani, L., Szummer, M., & Fitzgibbon, A. (2010). Efficient object category recognition using classemes. In *European Conference on Computer Vision (ECCV)* (pp. 776–789).

Tran, S. D., & Davis, L. S. (2008). Event modeling and recognition using markov logic networks. In *European Conference on Computer Vision (ECCV)* (pp. 610–623).

Vapnik, V., & Vashist, A. (2009). A new learning paradigm: Learning using privileged infromatin. *Neural Networks*, *22*, 544–557.

Vijayanarasimhan, S., & Grauman, K. (2008). Keywords to visual categories: Multiple-instance learning for weakly supervised object categorization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1–8).

Wang, H., & Schmid, C. (2013). Action recognition with improved trajectories. In *IEEE International Conference on Computer Vision (ICCV)* (pp. 3551–3558).

Wang, H., Klaser, A., Schmid, C., & Liu, C. L. (2011a). Action recognition by dense trajectories. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3169–3176).

Wang, L., Wang, Y., & Gao, W. (2011b). Mining layered grammar rules for action recognition. *International Journal of Computer Vision*, *93*(2), 162–182.

Xu, D., & Chang, S. F. (2008). Video event recognition using kernel methods with multilevel temporal alignment. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *30*(11), 1985–1997.

Yu, T. H., Kim, T.K., & Cipolla, R. (2010). Real-time action recognition by spatiotemporal semantic and structural forests. In *The British Machine Vision Conference (BMVC)* (p. 52.1–52.12).

Zeng, Z., & Ji, Q. (2010). Knowledge based activity recognition with dynamic bayesian network. In *European Conference on Computer Vision (ECCV)* (pp. 532–546).

Zhou, Z., & Zhang, M. (2006). Multi-instance multi-label learning with application to scene classification. In *Advances in neural information processing systems (NIPS)* (pp. 1609–1616).

Zhu, G., Yang, M., Yu, K., Xu, W., & Gong, Y. (2009). Detecting video events based on action recognition in complex scenes using spatio-temporal descriptor. In *Proceedings of the 17th ACM international conference on Multimedia* (pp. 165–174). ACM.